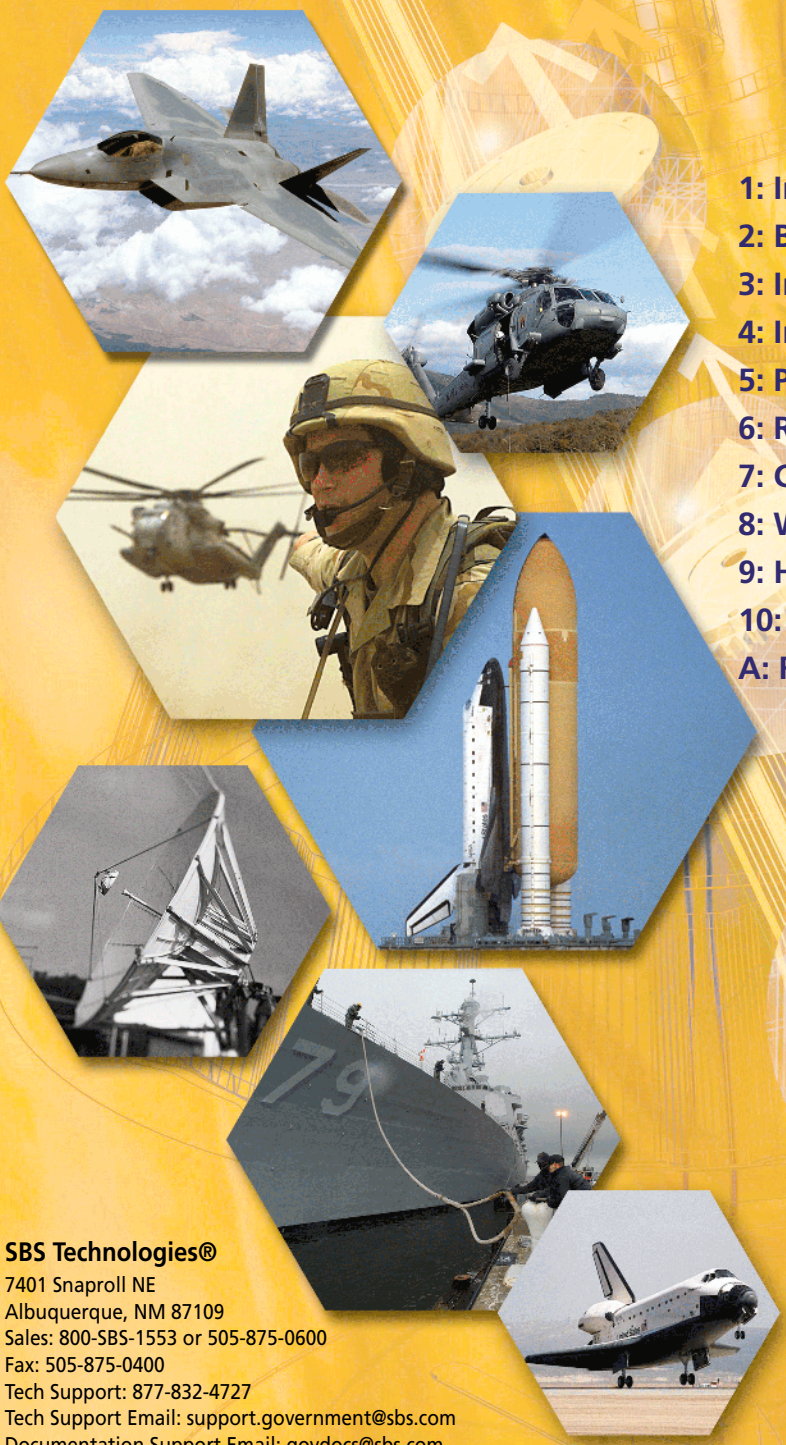
Getting Started with the ABI/ASF-PCI



SBS Technologies®
7401 Snaproll NE
Albuquerque, NM 87109
Sales: 800-SBS-1553 or 505-875-0600
Fax: 505-875-0400
Tech Support: 877-832-4727
Tech Support Email: support.government@sbs.com
Documentation Support Email: govdocs@sbs.com
http://www.sbs.com
http://www.resource.sbs.com

Applies to models:
ABI-PCI-1, ABI-PCI66-1, ASF-PCI-1, ASF-PCI66-1,
ABI-PCI-2, ABI-PCI66-2, ASF-PCI-2, ASF-PCI66-2

Doc PN: 504-553057-00

SBS Technologies®

Getting Started with the ABI/ASF-PCI

SBS Technologies, Inc., 7401 Snaproll NE, Albuquerque, NM 87109

1: Introduction

This chapter describes the contents of this manual, and the terminology and conventions used in this manual. This chapter contains the following sections:

Cross Reference: Appendix A gives a brief summary of technical revisions made to this manual.

1.1 Manual Overview

This manual applies to the following model numbers (the last digit in the model number indicates the number of channels):

- ABI-PCI-1
- ABI-PCI66-1
- ASF-PCI-1
- ASF-PCI66-1
- ABI-PCI-2
- ABI-PCI66-2
- ASF-PCI-2
- ASF-PCI66-2

The ABI/ASF-PCI cards run at 33 MHz. The ABI/ASF-PCI66 cards run at 33 or 66 MHz.

Use this manual to assist you in getting the ABI-PCI or ASF-PCI up and running as quickly as possible. It addresses the following:

- Installing the hardware
- Installing the software
- Configuring the software for your operating system
- Testing
- Basic module operation

This manual assumes that you will be using the SBS Integrated Avionics Library to operate the module. See Section 10.4 for instructions on starting up the module without using the library.

1.2 Contents of *MIL-STD-1553 ABI/ASF User's Manual*

The accompanying *MIL-STD-1553 ABI/ASF User's Manual* contains complete details on module programming and operation, including the following:

- MIL-STD-1553 programming and structures (Chapters 4–15)
- Sample 1553 applications (Chapter 16)

1.3 Contents of the *Integrated Avionics Library Reference Manual*

The *Integrated Avionics Library Reference Manual* provides information on using the included avionics libraries in your own application.

1.4 Terminology

Table 1.4.1 defines some of the basic terms used throughout this manual.

Table 1.4.1: Basic Terminology

Term	Meaning
BC	Bus controller
BM	Bus monitor
Bus	A single 1553 bus connection (i.e., Bus A or Bus B)
Channel	One complete, dual-redundant 1553 bus interface
Device	A logical entity that corresponds one-for-one with a 1553 channel and a device entry in the *sbs_dev.cfg* configuration file
Dual-redundant	Includes both a primary and a secondary connection (i.e., Bus A and Bus B make up a dual-redundant bus)
Firmware	Program running in the ABI/ASF digital signal processors that controls all 1553 operations. The firmware must be loaded upon device initialization.
RT	Remote terminal
SA	Subaddress
Word	A 16-bit value; i.e., two bytes

1.5 Conventions

The following conventions appear in this document. These conventions may differ from those used in other SBS publications. The subsections listed below describe each convention in more detail:

- Typographic Conventions
- Words Having Special Meaning
- Compound Keystrokes and Menu Selections
- Symbols

1.5.1 Typographic Conventions

Table 1.5.1 shows the typographic conventions used in this document.

Table 1.5.1: Typographic Conventions

Element	Use in Body Text	Use in Procedures
Italic	➢ Cross references to other SBS publications ➢ Filenames and directory paths ➢ Emphasis	➢ Cross references to other SBS publications ➢ Filenames and directory paths
Bold	➢ (Not used in body text)	➢ Controls, dialogs, menus, and text or numeric fields that appear on the screen ➢ Keys on your keyboard
`Courier Roman`	➢ Code examples	➢ Simulating the appearance of screens
`Courier Bold`	➢ Library function calls and syntax ➢ Emphasizing lines of code	➢ Commands and other information that you type as given
Angle brackets, e.g.,< >	➢ Enclosing variable information that you type (without the brackets) in place of a dummy variable	➢ Enclosing variable information that you type (without the brackets) in place of a dummy variable

The point size of the text varies depending on whether it is used in body text, code examples, notes, screens, or procedures.

1.5.2 Words Having Special Meaning

In procedures, the words “Enter” (or “enter”) and “Type” (or “type”) have special meanings that are indicated in Table 1.5.2.

Table 1.5.2: Words with Special Meaning

Word	Meaning
Enter	Key in the specified text or variable information and press the **Return** key.
Type	Key in the specified text. Do not press **Return**.

1.5.3 Compound Keystrokes and Menu Selections

Compound Keystrokes Whenever a procedure instructs you to press multiple keys, a double angle bracket “»” separates the names of the keys. Table 1.5.3 shows an example.

Menu Selections Whenever a procedure instructs you to select an item from a pull-down menu, a double angle bracket “»”separates the menu items. Table 1.5.3 shows an example.

Table 1.5.3: Examples of Notation for Compound Keystrokes and Menu Selections

Instruction	Meaning
Press **Ctrl** » **Alt** » **Delete**.	Press the **Ctrl**, **Alt**, and **Delete** keys simultaneously.
Select **File** » **Open**.	Select **Open** from the **File** menu.

1.5.4 Symbols

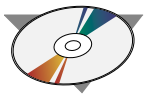
The following symbols appear throughout this manual:

Warning: Paragraphs next to this symbol contain information critical to module operation or to your safety.

Note: Paragraphs next to this symbol contain information important to module operation.

Tip: Paragraphs next to this symbol contain useful tips.

Cross Reference: Paragraphs next to this symbol contain cross references to other parts of this manual, or to other SBS publications.



Software Cross Reference: Paragraphs next to this symbol contain cross references to software media included with this product.

2: Before You Begin

The sections in this chapter describe what to do after receiving and prior to installing your card. This chapter contains the following sections:

2.1 What You Should Have Received

- ABI-PCI-1, ABI-PCI66-1, ASF-PCI-1, ASF-PCI66-1, ABI-PCI-2, ABI-PCI66-2, ASF-PCI-2, or ASF-PCI66-2 interface module
- Cable assembly
- SBS Resource CD - Contains PDF versions of this manual, the *MIL-STD-1553 ABI/ASF User's Manual,* and the *Integrated Avionics Library Reference Manual*

2.2 Unpacking the Card

Warning: This is an electronic product that is sensitive to electrostatic discharge. Take normal precautions in handling the card to prevent damage.

- Carefully unpack the card and inspect it for physical damage that might have occurred during shipping.
- If you have a damaged card, contact the SBS technical support group that handles maintenance, repairs, and warranties in Albuquerque. When you call us, give us the serial number of your card, and have the card available in case we have questions about its condition. You can find the serial number on a white tag on the card.

2.3 What You Will Need

The ABI/ASF-PCI product package includes all items required to operate the card on your chassis except for the following:

For a Single-Device PCI

- Two MIL-STD-1553 bus terminators, to perform a bus test on the module
- Two single bus couplers or other appropriate transformer coupling devices, to connect to an actual 1553 bus

For a Dual-Device PCI

- Four MIL-STD-1553 bus terminators, to perform a bus test on the module
- Two dual or four single bus couplers or other appropriate transformer coupling devices, to connect to an actual 1553 bus

Cross Reference: See Subsection 9.2.5 for ordering information and part numbers for these items.

3: Installing the Software

The sections in this chapter include descriptions of the provided software disks and instructions on software installation. This chapter contains the following sections:

3.1 Support Software

SBS provides support software for its MIL-STD-1553 products as part of the Integrated Avionics Library on the following media:

- SBS Resource CD

SBS Resource CD

The SBS Resource CD contains the following:

- Integrated Avionics Library, including C library source files, DLLs, sample applications, and the console mode version of the Unit Test executable
- Device drivers necessary to support the interface between the libraries and your computer system
- SBS PASS demo software
- Product documentation in PDF format (requires Adobe Acrobat Reader)
- Firmware files that have to be downloaded to the PCI card upon initialization

3.2 Copying the Software to Your Host System

Use the following instructions to copy the software to the system in which you are installing the PCI card. The following two sections include separate instructions for Windows and UNIX systems:

- Windows Operating Systems
- UNIX-Based Operating Systems

3.2.1 Windows Operating Systems

Unless otherwise specified, the installation batch files will place the Integrated Avionics Library on your system. The *Integrated Avionics Library Reference Manual* provides information on using the library in your own application, and operating system dependent software in the *c:\sbs_ver<x.yz>* folder (directory) in your Windows 95/98/ME/2000 or Windows NT environment, where *<x.yz>* is the version number of the current release.

1. Install the library by completing the following steps:

   - Insert the SBS Resource CD into your CD-ROM drive.
   - Start the Explorer and navigate to the CD.
   - Double-click on the Library folder.
   - Double-click **setup.exe** from the Explorer.
   - Follow the instructions that appear on the screen.

   If you select all of the defaults, it creates the directory structure shown in Figure 3.2.1.

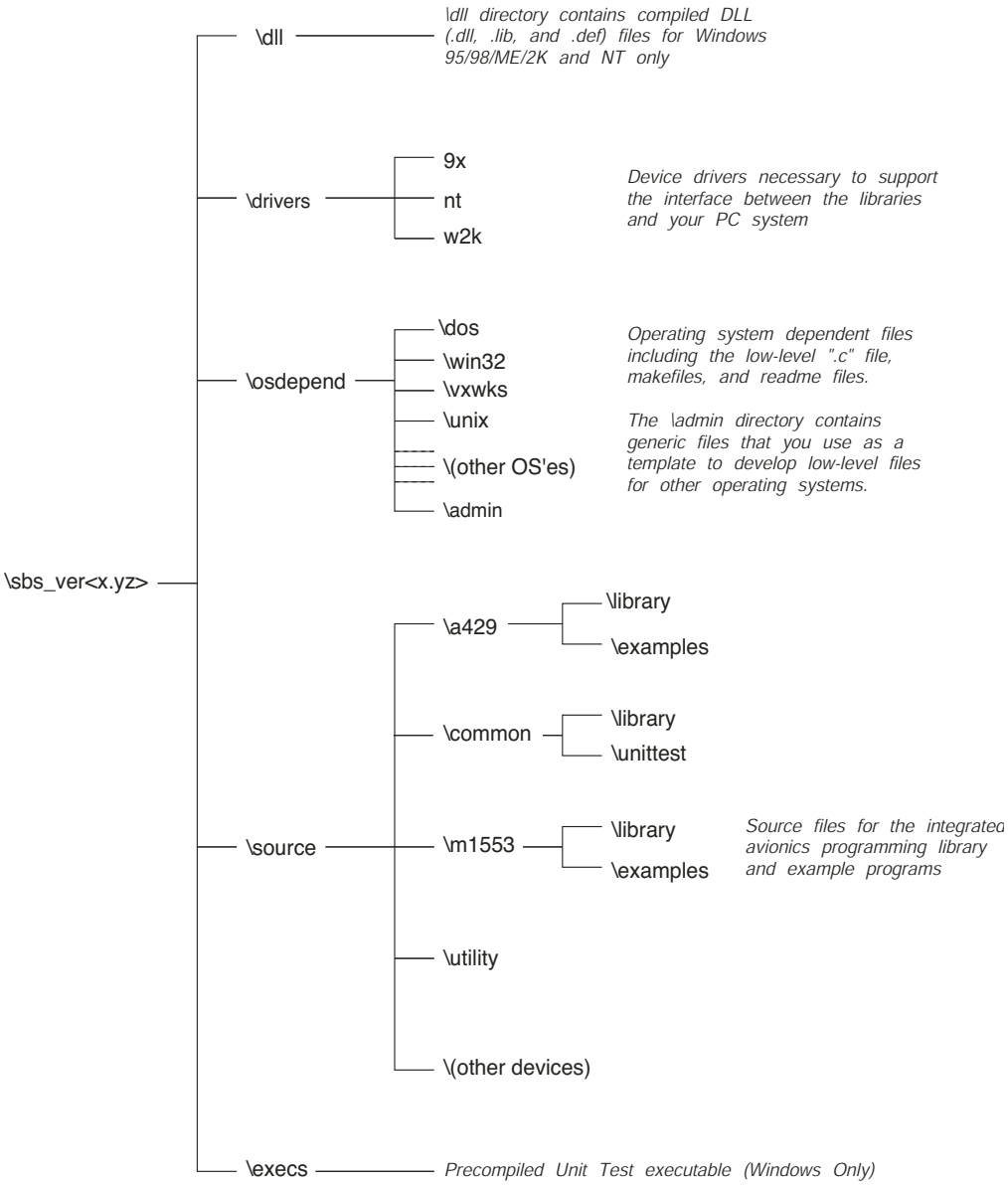


Figure 3.2.1: Default Directory Structure

2. Create the SBS user directory structure shown in Figure 3.2.2 by adding *working* and *firmware* directories under the *sbs_ver<x.yz>* directory, where *<x.yz>* is the version number of the current release.

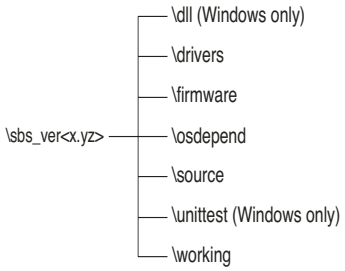


Figure 3.2.2: User Directory Structure

Note: The *\working* directory is a place for you to create and work with files without disturbing other files and directories.

3. Copy the firmware file(s) directly from the Firmware disk to the *\sbs_ver<x.yz>\firmware* directory on your system.

3.2.2 UNIX-Based Operating Systems

To install the interface libraries and operating system software on your UNIX system, you must first copy the software to a PC system on your network, then transfer it via ftp to the UNIX system in which you are installing the PCI card. The steps below guide you through the process.

1. Copy the software to a PC system on your network by following procedure steps 1–3 beginning on page 10 in the preceding subsection, Windows Operating Systems. Make sure to select the files needed for your target UNIX system, not the PC system.

2. Enter **`ftp`** to access your host system.

3. Create the directory structure shown in Figure 3.2.3 using the ftp **`mkdir`** utility:

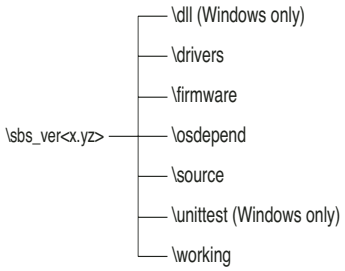


Figure 3.2.3: User Directory Structure

4. Use the ftp **`put`** or **`mput`** utility to transfer the interface library files from the PC environment to the host system (into the above tree structure).

4: Installing the Hardware

The sections in this chapter provide instructions on how to configure and install the ABI/ASF-PCI or ABI/ASF-PCI66 hardware. This chapter contains the following sections:

4.1 Installing the Card

To install the PCI card, follow the procedure below.

1. Insert the PCI card into one of the PCI slots in your system.
2. Power up the host system.

Figure 4.1.1 and Figure 4.1.2 show a top and rear panel view of the single-device PCI card. Figure 4.1.3 and Figure 4.1.4 show a top and rear panel view of the dual-device PCI card.

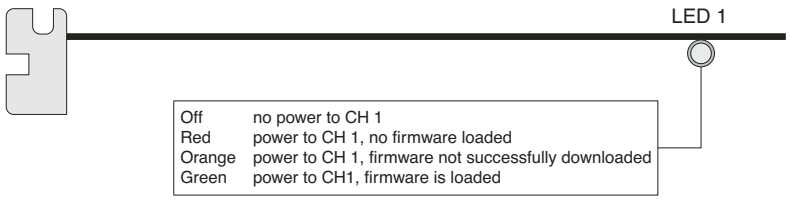


Figure 4.1.1: Single-Device PCI, Top View as Installed

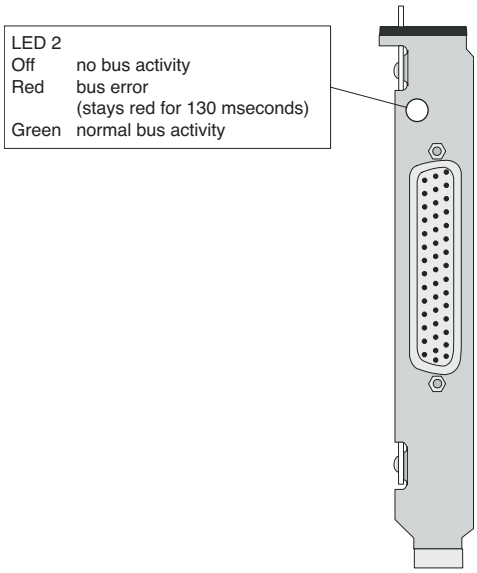


Figure 4.1.2: Single-Device PCI, Rear Panel LED

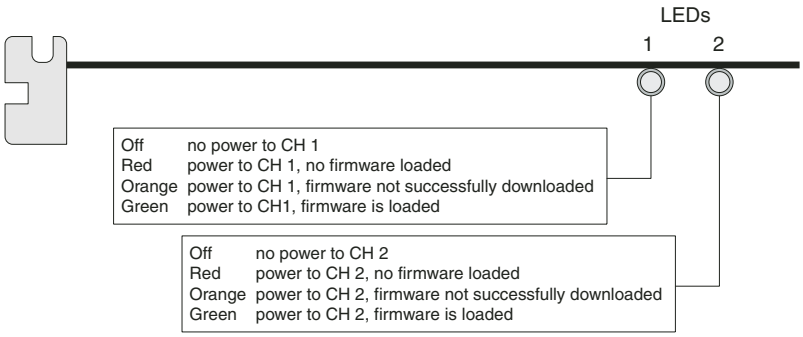


Figure 4.1.3: Dual-Device PCI, Top View as Installed

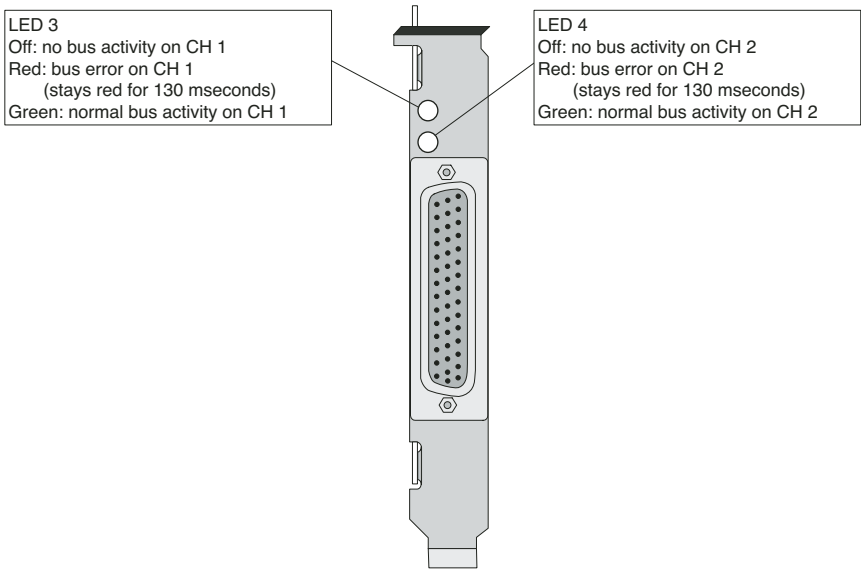


Figure 4.1.4: Dual-Device PCI, Rear Panel LEDs

4.2 Making Auxiliary Connections for External Signals

If you are planning to use external signals, you need to make the proper connections to the 15-pin connector shown in Figure 9.2.2 or Figure 9.2.4. See the instructions below to make the connections for IRIG signals and for external triggers.

IRIG Signal

If you are using an external IRIG signal as your timing source, make the following connections:

1. Connect the IRIG signal to pin 7 (IRIG In) of the 15-pin connector.
2. Connect the ground to pin 15 (GND) of the 15-pin connector.

Note: The IRIG input impedance is 10kΩ.

Cross Reference: For more information on IRIG signals, see the Device Management Programing Chapter of the *ABI/ASF User's Manual*.

External Trigger

If you are using an external trigger (in or out), make the following connections:

1. Connect the trigger signal to pin 4 of the 15-pin connector.
2. Connect the ground to pin 15 (GND) of the 15-pin connector.

Cross Reference: For more information on external triggers, see the Device Management Programing Chapter of the *ABI/ASF User's Manual*.

4.3 Attaching the Cable Assembly

Testing and actual 1553 operation of the card requires the use of the cable assembly included with the PCI module. It provides leads to attach the PCI to the 1553 bus.

Before Testing the PCI

1. Attach the cable assembly to the J1 connector on the rear panel of the card.
2. Attach 1553 bus terminators to each Channel lead on the cable assembly, as illustrated in Figure 4.3.1 for the single-device PCI and Figure 4.3.2 for the dual-device PCI.

Cross Reference: Before operating the PCI on a 1553 bus, see the instructions on page 43 of this manual for connecting the PCI to a 1553 bus.

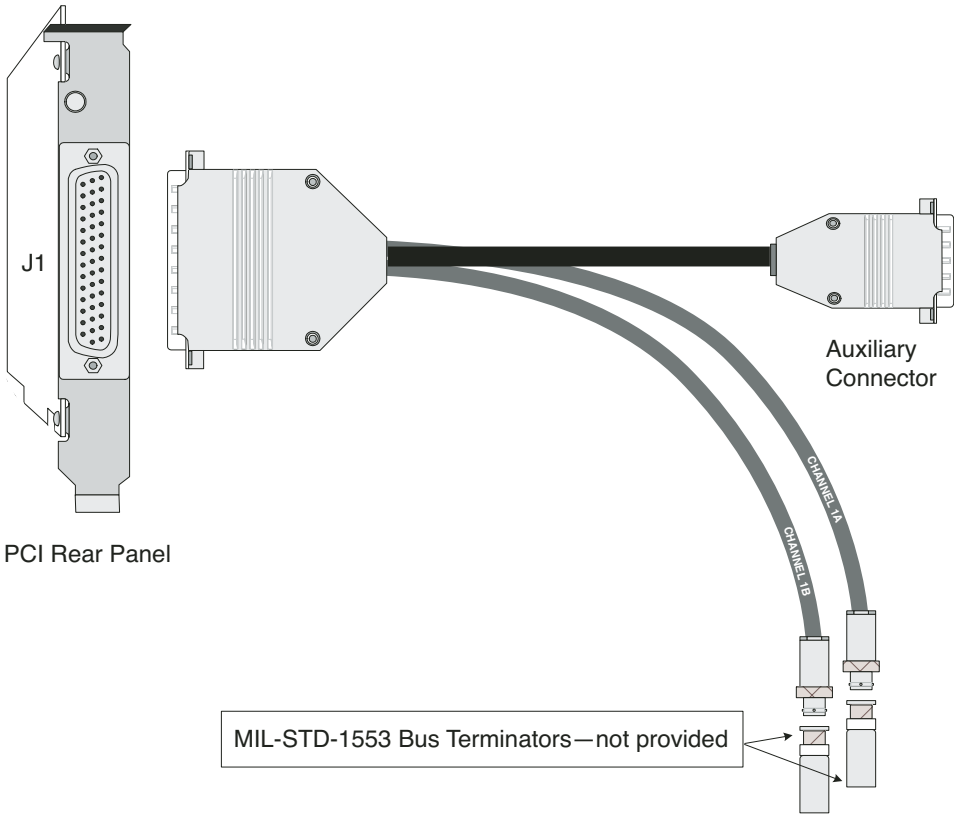


Figure 4.3.1: Attaching the Cable Assembly to the Single-Device PCI

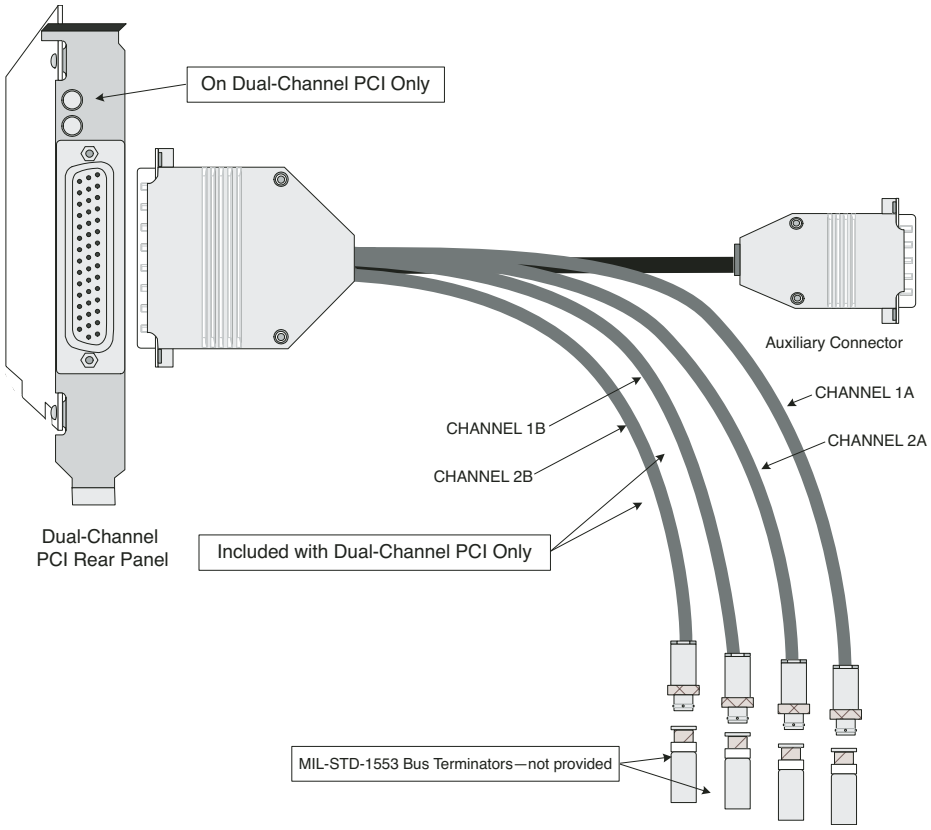


Figure 4.3.2: Attaching the Cable Assembly to the Dual-Device PCI

5: Performing System-Specific Installation

Note: These procedures are for use with non-PASS cards. For PASS card installation, refer to the *PASS 3200 User's Manual*.

The sections in this chapter provide instructions on how to install the files specific to your operating system and platform. This chapter provides instructions for the following systems:

- Windows 95/98/ME
- Windows NT
- Windows 2000
- IRIX 6.5

Note: If you are using a system not covered in this chapter, see Chapter 21 of the *Integrated Avionics Library Reference Manual* for instructions on installing the files for your system.

5.1 Windows 95/98/ME

Use this procedure with the Integrated Avionics Library, Version 6.3X or later, where PASS-3200 has not been installed.

1. After you have inserted the card into one of the slots in your system and powered up your system, the **New Hardware Found** dialog appears. Select **Driver from disk provided by hardware manufacturer** and click **OK**.
2. Click the **Browse** button.
3. Navigate to the drive where you installed the SBS software.

Note: If you're using Windows 95, the operating system copies the *.vxd* and the *.inf* files automatically.

4. Select *sbs_ver<x.yz>\drivers\sbs.inf*, where *<x.yz>* is the version number of the current release, and click **OK**.
5. If you are prompted to specify the location of the virtual device driver *sbspci.vxd*, click the **Browse** button, select *sbs_ver<x.yz>\drivers\sbspci.vxd*, and click **OK**.

Note: The computer configures parameters such as I/O base, base memory address, and interrupt vector. The *sbs_dev.cfg* (or *dev_cfg.h*) configuration file ignores these parameters.

5.2 Windows NT

Use this procedure with the Integrated Avionics Library, Version 6.3X or later, where PASS-3200 has not been installed.

Warning: In order to run **configmgr**, you must have Microsoft Internet Explorer version 4.0 or higher installed on your system.

Note: You must log on with Administrator privileges before performing these procedures.

1. If you're using the Windows NT operating system, copy the files in the *[drive]:\sbs_ver<x.yz>\drivers\Windows NT* directory, where *<x.yz>* is the version number of the current release, to the *[drive]:\winnt\system32\driver* directory.

2. Execute *[drive]:\sbs_ver<x.yz>\source\utility\configMgr.exe*.

Warning: Do not execute this program from the SBS Resource CD.

3. Select the **ADD PC Board** menu option.

4. Select **PCI Board** from the four board types offered. Once you select the type, follow the appropriate instructions for entering board-specific information.

   For PCI Boards:

   - Select either **1553 Single Device** or **1553 Dual Device** as appropriate for the board type from the **Board Type** pull down menu.
   - Select the appropriate board function mode by clicking on the **Mode** radio button under **Board Type**. For 1553 boards, select **ASF** (single function) or **ABI** (multi function).
   - Click the **Apply** button, and proceed to Step 5.

5. Repeat Step 3 and Step 4 for any additional boards.

6. Click on the **Restart driver to update changes** button.

7. When finished, reboot for the changes to take effect.

5.3 Windows 2000

Use this procedure with the Integrated Avionics Library, Version 6.3X or later, where PASS-3200 has not been installed.

Cross Reference: To uninstall the Windows 2000 SBS drivers, see the instructions on page 24 of this manual for Uninstalling SBS Drivers In Windows 2000.

Warning: In order to run **configmgr**, you must have Microsoft Internet Explorer version 4.0 or higher installed on your system.

Note: You must log on with Administrator privileges before performing these procedures.

1. Install your SBS device in your unplugged system.
2. After your SBS device is installed, plug in your system and start Windows 2000.

   The system indicates it found the device and the **Welcome to the Found New Hardware Wizard** window appears.
3. Select **Next**.

   The **Install Hardware Device Drivers** window appears.
4. Select the bottom choice, **Display a list** and click on **Next**.

   The **Hardware Type** window appears.
5. Scroll down to the icon called **Other devices**. This has a (yellow) question mark (?) in front of it. Click **Next**.

   The **Select a Device Driver** window appears.
6. Select the button labeled **Have Disk**.

   The **Install From Disk** window appears.
7. In the lower right portion of the window, select **Browse**.
8. Locate your SBS driver directory for Windows 2000. Select *SBSWDMPC.inf* for the PCI card. Select **Open**.

   The **Install From Disk** window reappears.
9. Click the button labeled **OK**.

   The **Select a Device Driver** window appears.
10. Select your device from the list. For example, the PCI version of the WMUX is

the choice **SBS 1553 PCI (PLX 9080)**. Select **Next**.

Note: If you get a warning that it cannot be verified, disregard and close this warning window.

The **Start Device Driver Installation** window appears.

11. Select **Next**.

    The files needed are copied.

    The **Completing the Found New Hardware Wizard** window appears.

12. Select **Finish**.

13. Now, execute *[drive]:\sbs_ver<x.yz>\source\utility\configMgr.exe*, where *<x.yz>* is the version number of the current release of your installed libraries. Click on the 1st blank line, select **Add PC Card** from the menu, then select and click the proper card choice (**PCI Board**).

    Table 5.3.1 lists the windows which can be displayed for the PC card selection choice.

Table 5.3.1: PC Card Selection Choice and Window Displayed

Add PC Card Menu Choice	Window Displayed
ISA Board	PC3 Card Setup
PCI Board	PCI Card Setup
PCMCIA Board	PCMCIA Card Setup
PC104 Board	Setup 1553-PC104

14. In the next window, select either **1553 Single Device** or **1553 Dual Device** as appropriate for the board type from the drop-down menu and either **ABI** or **ASF**.

15. Restart your computer for the change to take effect.

16. Setup your *SBS_DEV.cfg* file to reflect the device you have. Place this copy, the *unittest.exe* file, and your firmware file (*F066J.DAT* for the WMUX or similar for other cards) in the appropriate directory.

    You should now be able to use your *Unittest.exe* to verify device setup and operation.

Uninstalling SBS Drivers In Windows 2000

Use this procedure with the Integrated Avionics Library, Version 6.3X or later, where PASS-3200 has not been installed.

Note: You must log on with Administrator privileges before performing these procedures.

1. Start Windows 2000.
2. Open the **Control Panel**. Double click the **Add/Remove Hardware** icon.
3. Click **Next** in the 1st window. Once the 2nd window appears, select the bottom choice, **Uninstall/Unplug a device**. Click **Next** to go to the next screen.

   The **Choose a Hardware Device** dialog appears.
4. Scroll down the list until you find the SBS device in the list. Highlight it, then select **Next**.

   The **Completing the Add/Remove Hardware Wizard** window appears.
5. Select **Finish** to complete this stage of the uninstallation.

Note: Windows 2000 might start a hardware troubleshooter help window at this point. This window is not needed; so, close it.

6. In **Explorer**, use browse to find your *Inf* directory. Generally, the location of this directory is *c:\WINNT\Inf*.
7. Look at the files *OEM<X>.inf*, where *<X>* is a number starting at zero (0). There could be 10 or more files. Double click on the file(s) to open them. (If Explorer asks you what to open the file with, use the **Notepad** application. Be sure to check the **Always use this program** checkbox).
8. The top line of the file indicates if it is SBS file or not. Look for the PCI driver which states: *;INF file for SBSWDMPCI32.sys*. After finding *;INF file for SBSWDMPCI32.sys* file, rename or delete this file. Also, rename or delete the associated *OEM<X>.PNF* file.

Note: This file may or may not be in your system, depending on how it was originally installed.

9. (Optional) Search for and rename or remove all SBS drivers in your *c:\WINNT\SYSTEM32*, *c:\WINNT\SYSTEM32\Drivers*. These drivers are prefixed with SBS. Starting in your c:\WINNT directory, search for *SBS*.VXD* and *SBS*.SYS*. Search should locate all of these files.
10. Restart the system to clean out the rest of the files that may be in use.

Note: If you have a PCI device, completely power down the system for at least 5 seconds.

11. Turn off the system and remove the card unless you are going to reinstall the driver.

5.4 IRIX 6.5

In order to operate the PCI card in a SGI machine that utilizes IRIX 6.5.X, you need to be able to open the assigned UNIX system path. Each time you boot the machine with the **–r** option, it recreates this path. If you move or add/remove cards, the pathnames of the cards will likely change. This could result in having to modify the *sbs_dev.cfg* file to enter the new information. The following steps provide a procedure to identify the card and compile the unit test code.

1. Install the card into the card cage and reboot the system using the **–r** option to force the IRIX system to redetect the cards installed. On some systems the **–r** is not required. If you're unsure, check with your system administrator.

2. Login to the system.

3. Change directories to */hw/.id/pci* and do a **ls** command. The screen will display something similar to the following.

   ```
   10B5906D      11720C14      10B59080
   ```

   The numbers above are the Vendor/Device ID values for each PCI card installed in the system. For example, the above configuration shows two SBS cards installed. The 10B5906D would be for a single channel 1553 card and the 10B59080 would be either a Dual Channel 1553 card, or an 8 channel ARINC card.

4. Do an **ls –l** on each of these directories and record the symbolic link to the card. This is a base path to the card and will be different depending on your configuration. One example of the path appears below:

   /hw/module/1/slot/io2/pci_xio/pci/0

   This is the path that the system established to the card.

5. To access the user memory on the card, use an open command in the libraries to the Base Address 2 register of the PCI interface chip. The path to this register is */base/2* added to the path above and results in the following path.

   /hw/module/1/slot/io2/pci_xio/pci/0/base/2

   Note this path and record it for later use. The **`dd_name= field`** of the *sbs_dev.cfg* file uses this path.

6. Repeat step 5 for each SBS card in the system.

6: Running Unit Test

To verify that the PCI is properly installed and operational, use the instructions in the following sections to run the Unit Test application:

- Introduction
- Using the Combined (1553, A429, and WMUX) Unit Test Executable
- Setting up the Device Configuration File
- Unit Test Using the Console Mode
- Troubleshooting

Note: You must have an ANSI compatible terminal or driver in order to run Unit Test.

6.1 Introduction

The *\sbs_ver<x.yz>\execs* directory, where *<x.yz>* is the version number of the current release, supplies the executable for Windows for the combined Unit Test application. If you are using one of these operating systems with the PCI card and wish to use the combined Unit Test, proceed to the next section. If you do not have access to the distributed Unit Test executables, if there is not a pre-compiled executable for your operating system, or if you wish to run a stand-alone 1553 Unit Test, use the instructions in the *Compiling Your Application* Chapter of the *Integrated Avionics Library Reference Manual* to compile a new Unit Test executable.

You can begin using the Unit Test by going to Unit Test Using the Console Mode in Section 6.4.

6.2 Using the Combined (1553, A429, and WMUX) Unit Test Executable

This section describes the procedures for using the combined Unit Test. The topics are as follows:

- Operating Systems with File Systems
- Operating Systems without File Systems

6.2.1 Operating Systems with File Systems

To use the precompiled, combined Unit Test for Windows, complete the following steps:

1. Copy the Unit Test executable from the *\sbs_ver<x.yz>\execs\unittest.exe* directory to the *\sbs_ver<x.yz>\working* directory (where *<x.yz>* is the version number of the current release).
2. Copy the *sbs_dev.cfg* file from the *\sbs_ver<x.yz>\source\common\library* directory to the *\sbs_ver<x.yz>\working* directory.
3. Copy the firmware files from the *\sbs_ver<x.yz>\firmware* directory to the *\sbs_ver<x.yz>\working* directory.
4. Set up the *sbs_dev.cfg* file as described in the next section, *Setting up the Device Configuration File*.

6.2.2 Operating Systems without File Systems

1. Refer to the Compiling Your Application Chapter of the *Integrated Avionics Library Reference Manual* to determine the necessary files and appropriate compiler directives. Copy the appropriate files into the *\sbs_ver<x.yz>\working* directory, where *<x.yz>* is the version number of the current release.
2. Set up the *dev_cfg.h* file as described in the next section, *Setting up the Device Configuration File*.
3. Compile your Unit Test application as described in the Compiling Your Application Chapter of the *Integrated Avionics Library Reference Manual*.

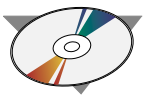
6.3 Setting up the Device Configuration File

You must define the device parameter values for your SBS device(s) in one of the following two files:

- *sbs_dev.cfg*, if your operating system has a file system
- *dev_cfg.h*, if your system does not have a file system

The *sbs_dev.cfg* device configuration file is an ASCII text file containing information that your application uses to initialize one or more SBS devices. The SBS device's initialization procedure calls a parser function (**`sbs_parse_config_file()`**) that reads the information in the configuration file. The parser function must be able to correctly read and verify the information for an SBS device before further library calls can be made. Because the **`sbs_parse_config_file()`** function reads the *sbs_dev.cfg* file at run-time, you can modify *sbs_dev.cfg* without recompiling the application.

If your operating system does not have a file system, you should use the *dev_cfg.h* file instead of the *sbs_dev.cfg* file. The *dev_cfg.h* file is an array of strings with the same format and keywords as the *sbs_dev.cfg* file. It parses in the same manner as the *sbs_dev.cfg* file. However, you must recompile your application each time you edit *dev_cfg.h* for your changes to take effect.



Software Reference: The location of *sbs_dev.cfg* and *dev_cfg.h* configuration files are in the directory in which you installed the Integrated Avionics Library under the *\install\library\source\common\library* directory.

The following sections provide a description and examples of the *sbs_dev.cfg* and *dev_cfg.h* files:

- Format
- Keywords
- sbs_dev.cfg File
- dev_cfg.h File

6.3.1 Format

The format of the *sbs_dev.cfg* and *dev_cfg.h* files resembles an *.ini* file found on most PCs. Values contained in this file vary based on your SBS device(s), system hardware, and operating system. Every device requires a separate set of keyword entries in the configuration file. (Thus, each single-device card requires a single set of keyword entries, and each multidevice card requires multiple sets.) The file is set up as follows:

- An equal sign (=) links each keyword with a value.
- The variable **<num>** represents a numerical value.
- The variable **<name>** represents a case-insensitive character string.
- The parser recognizes the following characters appended to the value string:
  - **b** (binary)
  - **o** (octal)
  - **h** (hex)
  - **d** (decimal; the default if no character is present)
- A semicolon at the beginning of a line denotes a comment line, and the parser ignores the entire line.

6.3.2 Keywords

Table 6.3.1 describes the keywords required in the *sbs_dev.cfg* or *dev_cfg.h* file for configuring the PCI card.

Table 6.3.1: Required Configuration File Keywords for the PCI Card

Keyword=<Specifier>	Required for:	Description
[DEVICE=<num>]	All operating systems and platforms	This line is required before all other keywords for a device. <num> represents a unique number, starting at 1, that identifies the SBS device.
base_address=<num>	SGI IRIX systems	<num> specifies (in bytes) the location of the SBS device in physical address space. This location is system and device specific. Consult your system user's manual for available memory locations.
dd_name=<name>	Windows NT, 9x, 2000, ME, and XP; some UNIX systems (not required for vxWorks)	<name> specifies the UNIX or Windows device driver name. It contains a text string with the name of the device driver node associated with the SBS device. For UNIX, <name> is the filename of the driver in the */dev* directory. For IRIX 6.4/6.5 systems, <name> is the XIO pathname that points to the memory space of the PCI bus for your card. For Windows NT, this value must correspond exactly to the name of the instantiated driver for the device, which can be found in the device directory using the Device Manager. It is of the form `sbspci320`, `sbspci321`, etc. For Windows 95/98, the driver name is `SBSPCI`.
device_type=<name>	All operating systems and platforms	<name> specifies the type of SBS card being used. For a single-device PCI, set the value to `M1553_PCI_1`. For a dual-device set the value to `M1553_PCI_1` for the first device and M1553_PCI_2 for the second device.

Keyword=<Specifier>	Required for:	Description
`firmware=<name>`	All operating systems and platforms	`<name>` specifies the firmware filename. If you define `NO_FILE_SYSTEM` in *sbs_sys.h*, the firmware will be loaded from a data array found in *firmware.h*. The array is specified by the "`firmware=`" entry in the *dev_cfg.h* configuration file. You can generate the *firmware.h* file using the utility program *setup_fw*.c. Devices that have flash memory still require this field to reprogram the flash memory. **Note:** Verify that the firmware filename listed in the *sbs_dev.cfg* or *dev_cfg.h* file is the same as that of the firmware file provided with the software distribution.
`relative_position=<num>`	Windows 95/98	`<num>` is the sequence number of the PCI card as found in the PCI bus slots. The first board is 0, the second board is 1, etc.

6.3.3 *sbs_dev.cfg* File

Note: If your operating system does not have a file system, you must use the *dev_cfg.h* file instead of *sbs_dev.cfg*. See page 33 for information on *dev_cfg.h*.

If your operating system has a file system, define the device parameter values for each SBS device in the *sbs_dev.cfg* file. Modify *sbs_dev.cfg* as follows:

1. Change directories to the *\sbs_ver<x.yz>\working* directory, where *<x.yz>* is the version number of the current release, if you are not already there.
2. Edit the *sbs_dev.cfg* file and verify that the settings are correct for the device type, firmware file(s), and number of devices you are using.
3. Save the file.

Examples of the parameters used for the *sbs_dev.cfg* file appear on the following page for a dual-device ABI-PCI card running under the following operating systems:

- Windows 95/98/ME
- Windows 2000/NT
- IRIX 6.5

These examples show only the required parameters. In the actual *sbs_dev.cfg* file you would remove the comment (semicolon preceding the parameter). All remaining lines are left commented out (preceded by a semicolon). If you are using a single-device card, you should uncomment keywords only for device 1. If you are running your card under an operating system other than those in the examples, refer to Table 6.3.1 to determine the keywords what your operating system requires.

Cross Reference: For an example of the complete *sbs_dev.cfg* file, refer to the *Integrated Avionics Library Reference Manual*.

Note: The values required for your system, including the firmware filename, may differ from those shown below.

Windows 95/98/ME

```
[DEVICE=1]
device_type=M1553_PCI_1
dd_name=sbspci
firmware=f025k.dat
[DEVICE=2]
device_type=M1553_PCI_2
dd_name=sbspci
firmware=f025k.dat
```

Windows 2000/NT

```
[DEVICE=1]
device_type=M1553_PCI_1
dd_name=sbspci320
firmware=f025k.dat
[DEVICE=2]
device_type=M1553_PCI_2
dd_name=sbspci321
firmware=f025k.dat
```

IRIX 6.5

```
[DEVICE=1]
base_address=0h
device_type=M1553_PCI_1
dd_name=/hw/module/4/slot/io2/pci_xio/0/base/2
firmware=f025k.dat
[DEVICE=2]
base_address=0h
device_type=M1553_PCI_2
dd_name=/hw/module/4/slot/io2/pci_xio/0/base/2
firmware=f025k.dat
```

6.3.4 *dev_cfg.h* File

Note: If your operating system has a file system, you should use the *sbs_dev.cfg* file instead of *dev_cfg.h*. See page 31 for information on *sbs_dev.cfg*.

This file contains the device information to be used in lieu of *sbs_dev.cfg* for embedded systems that do not have a file system. Modify *dev_cfg.h* as follows:

1. Change directories to the *\sbs_ver<x.yz>\working* directory, where *<x.yz>* is the version number of the current release, if you are not already there.
2. Initialize the **`dev_cfg`** string array with the configuration parameters for each card. The keywords in *dev_cfg.h* are identical to those in *sbs_dev.cfg*; see Table 6.3.1 on page 30 for the keyword descriptions.
3. Save the file.
4. Refer to the Sample 1553 Applications Chapter of the *ABI/ASF User's Manual* for information on compiling your application.

An example of the parameters used for the *dev_cfg.h* file appears below for a dual-device ABI-PCI card running under vxWorks.

This example shows only the required parameters. In the actual *dev_cfg.h* file you would remove the comment (semicolon preceding the parameter and following the leading quotation mark). All remaining lines are left commented out (preceded by a semicolon after the leading quotation mark). If you are using a single-device card, you should uncomment keywords only for device 1. If you are running your card under an operating system other than the one in the example, refer to Table 6.3.1 to determine the keywords what your operating system requires.

Cross Reference: For an example of the complete *sbs_cfg.h* file, refer to the *Integrated Avionics Library Reference Manual*.

Note: The values required for your system, including the firmware filename, may differ from those shown below.

```
char *dev_cfg_array[] = {
"[DEVICE=1]",
"device_type=M1553_PCI_1",
"firmware=f040k.dat",
"[DEVICE=2]",
"device_type=M1553_PCI_2",
"firmware=f040k.dat"
};
```

6.4 Unit Test Using the Console Mode

This section contains some of the basic procedures using the Console Mode Unit Test. This section contains the following procedures:

- Starting Unit Test Using the Console Mode
- Opening the 1553 Device
- Running Built-in Tests (BITs)
- Initializing the 1553 Device
- Exiting Unit Test

6.4.1 Starting Unit Test Using the Console Mode

Tips:
Press the carriage return key (**Enter**) to complete menu selections and enter responses to prompts.

Press **Enter** to restore a Unit Test menu following a failure.

1. Execute the Unit Test application.

   As soon as you execute Unit Test, it parses the configuration file (either *sbs_dev.cfg* or *dev_cfg.h*). If your configuration file parses without error, the menu shown in Figure 6.4.1 appears.

```
                    SBS Technologies, Inc.
                M1553 Interface Library Unit Test
                  Version X.YZ    Build MMM DD YYYY

                    1 - Device Management Tools
                    2 - Bus Controller Tools
                    3 - Remote Terminal Tools
                    4 - Bus Monitor Tools

                    q - Quit Unit Test

                Selection ? >
```

Figure 6.4.1: M1553 Interface Library Unit Test Menu

Note: Where shown, X.YZ is the version number of the current release and MMM DD YYYY is the date the current build was compiled.

Note: If you are using the precompiled unit test, select **m** at Integrated Avionics Library Unit Test screen and press return. The *M1553 Interface Library Unit Test Menu as shown in* Figure 6.4.1 appears.

If a parser error occurs, an error screen may appear instead of the Avionics Interface Library Unit Test Menu. Figure 6.4.2 shows a sample parser error screen. Table 6.5.1 describes the common parser error messages that you may encounter.

```
                    SBS Technologies, Inc.
                Integrated Avionics Library Unit Test
                    Version X.YZ Build XX.YY.ZZ

                  Failure parsing configuration file!
                  Device #1 missing "firmware=" keyword.

                  'Q' to quit, return to reparse.

                  Selection ? >
```

Figure 6.4.2: Sample Parser Error Screen

2. If a parser error occurs, proceed to the troubleshooting procedures in Section 6.5.1.

   If no parser error occurs, the menu shown in Figure 6.4.1 appears.

6.4.2 Opening the 1553 Device

1. To select **Device Management Tools**, enter **1**.

   The menu shown in Figure 6.4.3 appears.

```
              1553 Device Management Tools                |               pg1
                                                          |
  1 - Init Device <Steps 2-7>   a - Get Device Clock      |    n - Next Pg->
  2 -  Open Device              b - Set Device Clock      |
  3 -  Load Firmware                                     |    p - Pick dev #
  4 -  Start Application        d - Display Error Tbl    |
  5 -  Init Chan or V7 Mem      e - Clear   Error Tbl    |    r - r/w Ram
  6 -  Init Interrupt Q                                  |
  7 -  Create BSM Buffers                                |    s - Start i/o
                                                         |    t - sTop  i/o
  9 - Set ASF Mode              i - Execute BIT          |
  0 - Get Device Info           j - Close Device         |    x - eXit <-Pg
                                                         |
--------------------------------------------------------------------------------
Selection? >



Messages: Device # defaulted to 1.
--------------------------------------------------------------------------------

                   Device #1:  M1553_xxx_1  is  CLOSED                fyyy.dat
```

Figure 6.4.3: 1553 Device Management Tools Pg1 Menu

Note: Where shown, xxx is the card type (PCI, cPCI, etc.) of the card installed and fyyy.dat is the firmware being used.

Note: The device defaults to the first device number in the group of devices (in this case device #1). To select an alternate device number, enter **p**. At the "Device Number? >" message, enter the appropriate device number. The message "Device number changed to X. Hit ENTER to continue." appears (X is the number of the selected device). Press Enter. The device number selected appears at the bottom of the 1553 Device Management Tools menu.

2. To select **Open Device**, enter **2**.

   The messages shown in Figure 6.4.4 appear.

```
--------------------------------------------------------------------------------
Selection: 2

          Device #1 opened.
Messages: Hit ENTER to continue.
--------------------------------------------------------------------------------
```

Figure 6.4.4: Open Device Messages

3. To continue, press **Enter**.

   The area between the dashed lines clears. The bottom line displays "Device #X: M1553_xxx_1 is OPEN" where X is the number of the device and xxx is the card type.

LEDs

- Single-device
  - If the card was just powered up, LED 1 should be red. Otherwise, LED 1 should be off.
  - LED 2 should be off.
- Dual-device
  - If the card was just powered up, LEDs 1 and 2 should be red. Otherwise, LED 1 should be off.
  - LEDs 3 and 4 should be off.

Cross Reference: See Figure 4.1.1 on page 13 through Figure 4.1.4 on page 14 for locations and descriptions of the LEDs.

6.4.3 Running Built-in Tests (BITs)

1. To select **Execute Built-In Tests**, enter **i**.

   If you are prompted to enter a firmware source, enter the appropriate number to load from flash, from file, or via driver (depending on the type of card and operating system you are using, you may see only one of these options at the prompt).

The messages and prompt shown in Figure 6.4.5 appear.

```
--------------------------------------------------------------------------------
Selection: i
For the built in test to work properly, a cable assembly with terminators
must be attached to the ABI/ASF card and the bus must be quiet.

Enter q to quit, or ENTER to start test.

Messages:
--------------------------------------------------------------------------------
```

Figure 6.4.5: Execute Built-in Tests Verification Screen

2. Verify that the cable assembly with terminators is attached to the ABI/ASF card (see page 16 for instructions).

3. Press **Enter**.

   The messages and prompt shown in Figure 6.4.6 appear.

```
--------------------------------------------------------------------------------
Selection: i
'0' = [FILE]; '1' = FLASH; '2' = DRIVER
Firmware source? >



Messages:
--------------------------------------------------------------------------------
```

Figure 6.4.6: Firmware Selection Screen

4. Enter the appropriate number to load from file, from flash, or via driver (depending on the type of card and operating system you are using, you may see only one of these options at the prompt) and press **Enter**.

   If the tests are successful, messages and prompts similar to Figure 6.4.7 will appear over a 10- to 12-second period.

```
--------------------------------------------------------------------------------
Selection: i
'0' = [FILE]; '1' = FLASH; '2' = DRIVER
Firmware source? > 1
Please Wait (Takes 6 to 9 seconds)...
           BIT passed!
           Device closed. Initialize before running.
Messages:  Hit ENTER to continue.
--------------------------------------------------------------------------------
```

Figure 6.4.7: Execute Built-in Tests Passed Screen

   If the tests are unsuccessful, error messages appear.

5. To continue (in either case), press **Enter**.

   The area between the dashed lines clears. The bottom line displays "Device #X: M1553_xxx_1 is CLOSED" where X is the number of the device and xxx is the card type.

LEDs

- Single-device
  - LED 1 should turn green. This indicates that the firmware was successfully loaded.
  - Next, LED 2 should turn green for a few seconds. This indicates that the bus is being tested.
  - Finally, LED 1 and then LED 2 should turn off.
- Dual-device
  - LEDs 1 and 2 should turn green. This indicates that the firmware was successfully loaded.
  - Next, LEDs 3 and 4 should turn green for a few seconds. This indicates that the bus is being tested.
  - Finally, LEDs 1 and 2 and then LEDs 3 and 4 should turn off.

Tips:
If the tests are unsuccessful, check for proper termination of the bus.

Contact SBS technical support if you are unable to correct the problem.

6.4.4 Initializing the 1553 Device

1. To select **Initialize Device**, enter **1**.

   The messages and prompts shown in Figure 6.4.8 appear one at a time.

```
------------------------------------------------------------------
Selection: 1

Interrupt queue length [4]? >
Seq. monitor length [1000]? >
'0' = [FILE]; '1' = FLASH; '2' = DRIVER
Firmware Source? >

Messages:
------------------------------------------------------------------
```

Figure 6.4.8: Initialize 1553 Device Prompts

2. To set the queue length to 4 entries, press **Enter** at the Queue Length prompt.
3. To set the sequential monitor length to 1000 words, press **Enter** at the Sequential Monitor Length prompt.
4. If you are prompted to enter a firmware source, enter the appropriate number to load from file, from flash, or via driver (depending on the type of card and operating system you are using, you may see only one of these options at the prompt).

If the initialization process completes successfully, the message "Device initialized" appears. If this process does not complete successfully, an error appears between the dashed lines.

5. Press **Enter** to continue.

   The area between the dashed lines clears. The bottom line displays "Device #X: M1553_xxx_1 is STOPPED" where X is the number of the device and xxx is the card type.

LEDs

- Single-device
  - LED 1 should turn green.
  - LED 2 should remain off.
- Dual-device
  - LEDs 1 and 2 should turn green.
  - LEDs 3 and 4 should remain off.

Cross Reference: See Figure 4.1.1 on page 13 through Figure 4.1.4 on page 14 for the locations and descriptions of the LEDs.

Troubleshooting If a failure occurs during initialization, proceed to Section 6.5.2.

6.4.5 Exiting Unit Test

1. To return to the M1553 Interface Library Unit Test menu, enter **x**.

   The screen clears, and the M1553 Interface Library Unit Test Menu appears.

2. To exit from the Unit Test application, enter **q**.

   The prompt shown in Figure 6.4.9 appears.

```
Are you sure you want to quit? ([y]/n) >
```

Figure 6.4.9: Prompt to Quit

Note: If you are using the precompiled unit test, the Integrated Avionics Library Unit Test screen appears next. Select **q** at Integrated Avionics Library Unit Test screen and press return. The prompt shown in Figure 6.4.9 appears.

3. Enter **y**.

6.5 Troubleshooting

This section discusses troubleshooting if a parser error occurs during the Unit Test using the Console Mode or if a failure occurs during initialization. This section contains the following procedures:

- Parser Error
- Initialization Error

6.5.1 Parser Error

Unit Test Using Console Mode

1. Identify parser error using Table 6.5.1.
2. Correct the error in the configuration file. If you are using *sbs_dev.cfg*, press **Return** from the Unit Test error screen to reparse. If you are using *dev_cfg.h*, exit and recompile your code. Repeat step 1 in Section 6.4.1.

Note: After the configuration file parses without error (i.e., once you see the menu shown in Figure 6.4.1), you are ready to continue.

Table 6.5.1: Parser Error Messages

Error Message	Diagnosis
`Failure parsing configuration file! Error opening <filename> file!`	The indicated file could not be opened.
`Failure parsing configuration file! <keyword>=<num> for device #xx is invalid.`	The parser found a value out of limits in the configuration file. Correct the line containing `<keyword>`.
`Failure parsing configuration file! Application does not support <name> devices!`	The application type `<name>_APP` (where `<name>` is the avionics bus type) is not defined in the *sbs_sys.h* file.
`Failure parsing configuration file! Device #xx missing "<keyword>=" keyword.`	The specified `<keyword>`, which is required for the specified device (`Device #xx`, where `xx` is in the range 1 to `SBS_MAX_DEV`), was not found.
`Failure parsing configuration file! No device defined in configuration file!`	The `Device` keyword was not found in the configuration file.
`Failure parsing configuration file! "<name>" is not a valid device type.`	The `<name>` specified for the `device_type` keyword is not valid. Choose a valid device type from the list included in the configuration file.
`Failure parsing configuration file! "<filename>" not defined in the firmware.h file.`	The `<filename>` specified for the `firmware` keyword in *dev_cfg.h* does not match any firmware filename in the *firmware.h* file.
`Failure parsing configuration file! "<keyword>=<num>" on line xx: duplicate entry.`	A value for the `<keyword>` specified on line `xx` has already been parsed for this device number.
`Failure parsing configuration file! Line number xx has more than 132 characters.`	The number of characters on line `xx` exceeds the maximum number that can be read.

Error Message	Diagnosis
Failure parsing configuration file! "Device=0" on line xx must be greater than 0.	The `Device` keyword must have a value greater than zero.
Failure parsing configuration file! "Device=<num>" on line xx exceeds SBS_MAX_DEV.	The value `<num>` is greater than the `SBS_MAX_DEV` value specified in the *dev_mgmt.h* file.
Failure parsing configuration file! "Device=<num>" is a noncontiguous <name> 2nd device.	The values of the `Device` keywords for the first and second devices of the indicated board (`<name>` = `PC3` or `PC16`) must be consecutive numbers.

6.5.2 Initialization Error

If a failure occurs during initialization, do the following:

1. Turn off the system, and then physically remove and reseat the card in the host computer.
2. Check all cable connections and verify that they are secure.
3. Try initializing the card again.
4. If a failure still occurs, refer to Table 6.5.2.

Table 6.5.2: Initialization Errors and Error Messages

Error	Diagnosis
Computer system locked up	Interrupt request (IRQ) level is in use. Specify a different IRQ level in the *sbs_dev.cfg* or *dev_cfg.h* file.
One of the following error messages appeared:	
`Initialize device failed!` `sbs_open_device(): The device driver failed to initialize`	The device driver was not started on reboot. Verify that the device is started in Windows NT or that the device driver is selected and loaded in Windows 95/98.
`Initialize device failed!` `sbs_open_device(): Shared memory fail`	The **base_address** specified in the *sbs_dev.cfg* or *dev_cfg.h* file is invalid or there is a problem with the operating system device driver. On UNIX systems, verify that the correct filename is specified for the **dd_name** keyword in *sbs_dev.cfg* or *dev_cfg.h*.
`Initialize device failed!` `sbs_load_ram(): Download error`	The **base_address** specified in the *sbs_dev.cfg* or *dev_cfg.h* file is invalid. For MS-DOS, Windows 3.x, and Windows 95/98/NT, verify that the required memory region is excluded in the *config.sys* file.
`Initialize device failed!` `sbs_load_ram(): File open error`	Your application could not open the firmware file specified in *sbs_dev.cfg* or *dev_cfg.h*. Verify that the firmware filename specified in *sbs_dev.cfg* or *dev_cfg.h* is correct and that the specified firmware file is in your *\working* directory.
`Initialize device failed!` `sbs_start_firmware(): Start firmware failure`	The firmware did not properly start. Verify that the proper firmware file(s) are specified in *sbs_dev.cfg* or *dev_cfg.h*.
`Open device failed!` `sbs_open_device(): Device probe fail`	It is possible that the PCI device driver may be conflicting with the resources of another device. Try changing the memory setting for the PCI device in system resources. Conflicts with video display adapters using ROM Shadowing have resulted in resource conflicts.

Cross Reference: For further help with initialization failures, see *Customer Support Services* in the Introduction Chapter of the *MIL-STD-1553ABI/ASF User's Manual*.

7: Connecting to the 1553 Bus

If the Unit Test procedures above completes without error, you may proceed with MIL-STD-1553 operations. However, you must first connect each channel of the card to an actual 1553 bus by completing the following procedure:

1. Remove the terminators from the Bus A and Bus B cables (these cables are marked "Channel 1A" and "Channel 1B," respectively).
2. Attach the Bus A cable to the stub connector on an appropriate transformer-coupled device (see Figure 7.0.1).
3. Attach the Bus B cable to the stub connector on an appropriate transformer-coupled device (see Figure 7.0.1).

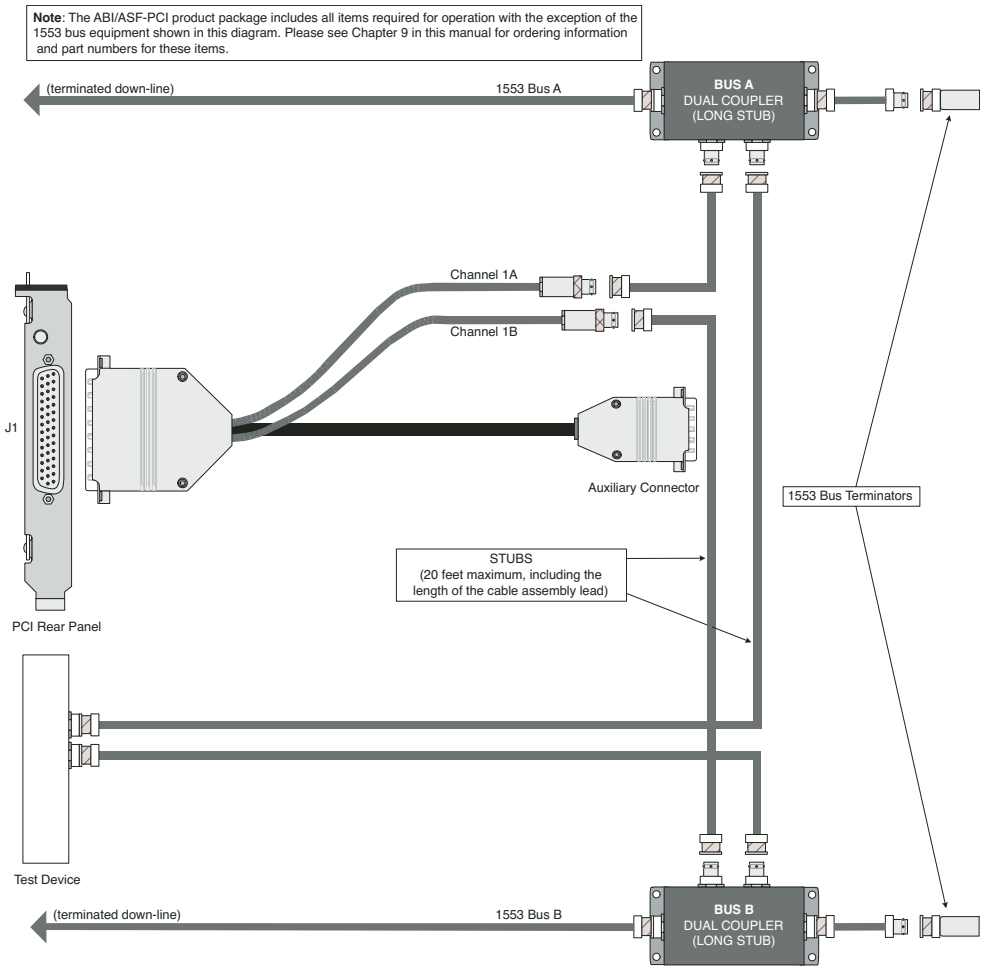


Figure 7.0.1: Connecting the PCI to a 1553 Bus

8: What to Do Next

If your card is properly connected to a 1553 bus, you may proceed in one or more of the following ways:

➢ Run sample 1553 applications.

Cross Reference: If you are a first-time user, SBS also recommends that you execute the sample 1553 applications that are included with the product shipment. See the Sample 1553 Applications Chapter of the *MIL-STD-1553 ABI/ASF User's Manual for t*hese sample applications.

➢ Build a custom 1553 application.

Cross Reference: If you are an advanced user, you may wish to begin designing your own 1553 application. See Chapters 4–15 in the *MIL-STD-1553 ABI/ASF User's Manual* for complete details.

9: Hardware Specifications

This chapter presents information about the ABI/ASF-PCI hardware, including:

- General Product Information
- Physical Specifications
- Operational Specifications

The General Product Information section contains general information for the ABI/ASF-PCI cards, including the cage code number, extended warranty information, conformance to MIL-STD-1553 electrical specifications, bus equipment part numbers and ordering instructions, and external signal characteristics.

In the Physical Specifications section, you can find information about component locations, dimensions, jumper and DIP switch settings, connectors, pinouts diagrams, and where you can order connecting cables and terminators.

The Operational Specifications section contains information about the operational aspects of the card, including temperatures, MTBF, and hardware reset.

9.1 General Product Information

The information in this section applies to the ABI/ASF-PCI hardware. This section contains the following topics:

9.1.1 Cage Code Number

0BAS8

9.1.2 Extended Warranty Information

SBS offers a comprehensive maintenance service for the ABI/ASF products. Even though SBS boards rarely fail, these services assure that the end user has thorough coverage and minimal down time in case of a failure.

SBS products include a two-year, parts and labor warranty. You can purchase an extended warranty to extend this service beyond the second year. This provides you with 10-day turn-around for the repair of a module (or provides a replacement module at no cost). Large-quantity repairs may require a longer turn-around time. The cost is minimal and agreements are normally for one-year periods.

For more information or to receive a copy of the maintenance agreement, contact SBS Technologies at one of the numbers listed on the inside cover of this manual and specify “Warranty Support.”

9.1.3 Conformance to MIL-STD-1553 Electrical Specifications

All SBS products conform to the 1553 electrical specifications illustrated in Figure 9.1.1.

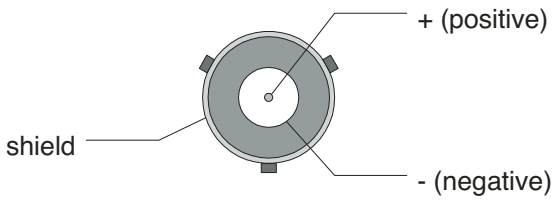


Figure 9.1.1: 1553 Electrical Specifications (as Viewed from End of Connector)

9.1.4 Bus Equipment Part Numbers and Ordering Instructions

You can purchase the bus equipment listed in Table 9.1.1 from SBS Technologies.

Table 9.1.1: OPE 1553 Bus Hardware Available from SBS Technologies

Part Number	Description
BUS-2	Dual-redundant bus with dual-stub couplers: four 15'-cables, four terminators, two 2-stub couplers
BUS-3	Dual-redundant bus with three-stub couplers: six 15'-cables, four terminators, two 3-stub couplers
Single Stub	Single-stub bus coupler
2 Stub	2-stub bus coupler
3 Stub	3-stub bus coupler
4 Stub	4-stub bus coupler
TERM	78-ohm terminator
CAB-COM-ZZ	Commercial-grade cable with PL-75 connectors (ZZ=length in feet)
CAB-MIL-ZZ	MIL17-grade cable with PL-75 connectors (ZZ=length in feet)
BUS-R	Regenerates bus signals for extending a MIL-STD-1553B Notice 2 compliant bus by 100 meters or 330 feet. Two BUS-R products are required for a dual-redundant bus.
BUS-C	Provides an RS-422 interface to SBS ABI modules to extend a MIL-STD-1553B stub from 20 to 300 feet
RS422-C	Cable Option: 300-ft RS-422 Cable for BUS-C

For more information, contact SBS at one of the numbers listed on the inside cover of this manual and specify "Sales Support."

Miscellaneous 1553 components are also available from the following vendor:

MilesTek
1506 Interstate 35 W
Denton, Texas 76207-2402
Attn: Al Stenzel
800-524-7444 or 940-484-9400
FAX: 940-484-9402

9.1.5 External Signal Characteristics

External Trigger

The external trigger feature is standard on the ABI/ASF-PCI. The external trigger is a transistor-transistor logic (TTL) signal having the characteristics shown in Figure 9.1.2.

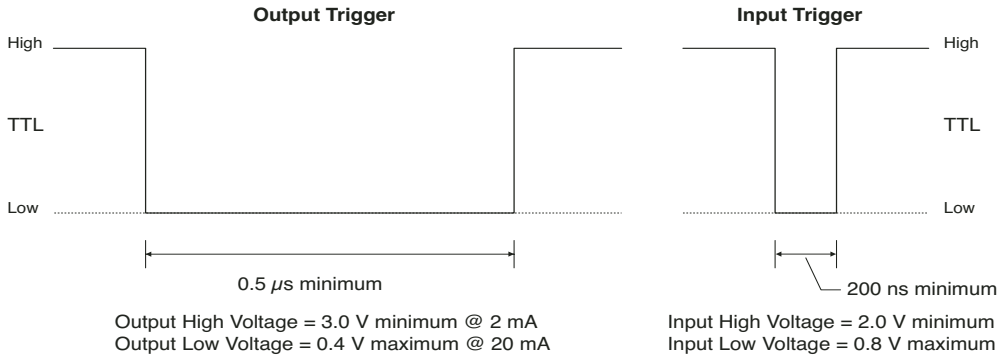


Figure 9.1.2: Characteristics of ABI/ASF External Trigger

Note: When the low or high level is selected for the input trigger, the signal must remain at that level for at least 20 microseconds. Otherwise, the firmware may not delete the input trigger.

External Clock

The ABI/ASF-PCI firmware supports the external clock feature. The external clock requires a differential signal with the specifications listed in Table 9.1.2.

Table 9.1.2: External Clock Differential Signal Specifications

Specification	Value
Recommended Differential Input Voltage (measured from + External Clock pin to – External Clock pin)	+12 V
Recommended Input Voltage (measured at either + pin or – pin)	Minimum: –7 V Maximum: +12 V
Differential Input Threshold	Minimum: –0.2 V Maximum: +0.2 V
Typical Output Voltages	High (minimum): +2.7 V Low (maximum): +0.5 V

IRIG

The IRIG clock feature is an option which must be ordered at the time of purchase. It is available on all ABI/ASF products. ABI/ASF products which include this option will accept an IRIG input signal compatible with the IRIG-B standard.

Note: The IRIG input impedance for all ABI/ASF products is 10 kΩ.

9.2 Physical Specifications

The information in this section describes the physical specifications for the ABI/ASF-PCI and the ABI/ASF-PCI66 cards. This section contains the following topics:

- Board Layout
- Board Dimensions
- Jumper Settings
- Connector Descriptions & Pinouts, and LED Functions
- Part Numbers and Ordering Instructions

9.2.1 Board Layout

The layout of the ABI/ASF-PCI-1 board is shown in Figure 9.2.1.

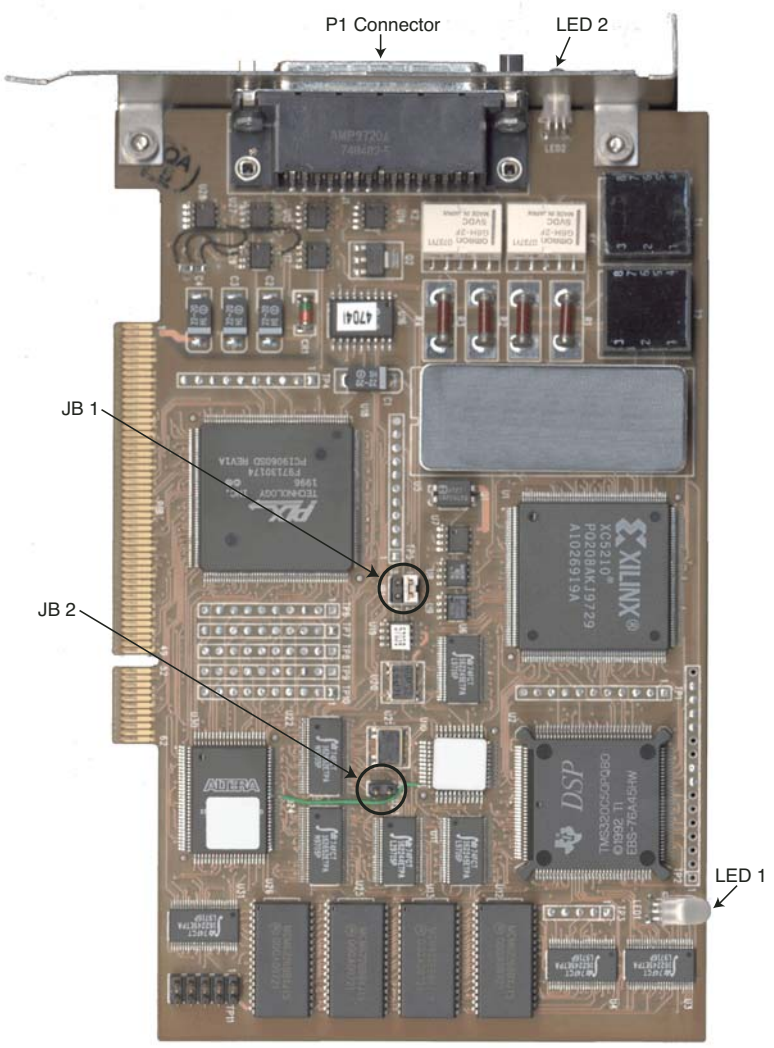


Figure 9.2.1: ABI/ASF-PCI-1 Board Layout

9.2.2 Board Dimensions

ABI/ASF-PCI-1: 175mm x 105mm
ABI/ASF-PCI66-1: 175mm x 105mm
ABI/ASF-PCI-2: 240mm x 105mm
ABI/ASF-PCI66-2: 240mm x 105mm

9.2.3 Jumper Settings

Table 9.2.1 describes the PCI jumper blocks. Upon power up, the setting of the JB2 jumper determines whether the firmware loads from a file, or loads from the built in PROM.

Table 9.2.1: PCI Jumper Settings

Jumper Block	Description	Default Jumper Setting
JB1	PCI serial EEPROM configuration	Pins 1–2 (jumper required for normal operation)
JB2	FLASH PROM autoload	Open (autoload disabled)

Note: These jumpers may not be present on all PCI boards.

9.2.4 Connector Descriptions & Pinouts, and LED Functions

Table 9.2.2 lists the pinouts for the J1 connector.

Table 9.2.2: PCI Pinouts for J1 Connector[a]

Pin	Standard Configuration	IRIG Option
1	Channel 1 A+	Channel 1 A+
2	Channel 1 B+	Channel 1 B+
3	Channel 2 A+	Channel 2 A+
4	Channel 2 B+	Channel 2 B+
5[b]	Ext Trg 1 +	Ext Trg 1 +
6[b]	Ext Trg 2 +	Ext Trg 2 +
7[b]	Ext Trg 3 *	Ext Trg 3 *
8[b]	Ext Trg 4 *	Ext Trg 4 *

* Active low signal
[a] References to channel 2 apply only to dual-channel PCI modules.
[b] Only External Trigger 3 is supported by firmware.

Pin	Standard Configuration	IRIG Option
9[b]	Ext Trg 5 *	Ext Trg 5 *
10	Ext Clock +	Ext Clock +
11	–	IRIG Input
12	–	–
13	Bias (1.7VDC)	Bias (1.7VDC)
14	+5VDC	+5VDC
15	+5VDC	+5VDC
16	Channel 1 A–	Channel 1 A–
17	Channel 1 B–	Channel 1 B–
18	Channel 2 A–	Channel 2 A–
19	Channel 2 B–	Channel 2 B–
20[b]	Ext Trg 1 –	Ext Trg 1 –
21[b]	Ext Trg 2 –	Ext Trg 2 –
22	–	–
23	–	–
24	–	–
25	Ext Clock –	Ext Clock –
26	GND (Ext Trg)	GND (IRIG/Ext Trg)
27	GND	GND
28	GND	GND
29	–	–
30–44	–	–

* Active low signal
[a] References to channel 2 apply only to dual-channel PCI modules.
[b] Only External Trigger 3 is supported by firmware.

The PCI ships with a cable assembly which attaches to the J1 connector. This assembly includes a cable lead with a 15-pin connector for making connections to auxiliary signals. Table 9.2.3 lists the pinouts for this 15-pin connector.

Table 9.2.3: PCI Pinouts for 15-Pin Connector on Cable Assembly

Pin	Corresponding Pin on J1	Signal
1	10	Ext Clock +
2[a]	5	Ext Trg 1+
3[a]	6	Ext Trg 2+
4[a]	7	Ext Trg 3*
5[a]	8	Ext Trg 4*
6[a]	9	Ext Trg 5*

* Active low signal
[a] Only External Trigger 3 is supported by firmware.

Pin	Corresponding Pin on J1	Signal
7	11	IRIG In
8	13	Bias
9	25	Ext Clock –
10[a]	20	Ext Trg 1–
11[a]	21	Ext Trg 2–
12	22	–
13	23	–
14	24	–
15	26	Gnd (IRIG/Ext Trg)

* Active low signal
[a] Only External Trigger 3 is supported by firmware.

Figure 9.2.2 and Figure 9.2.3 show the cable assembly and LEDs for the single-channel PCI card. Figure 9.2.4 and Figure 9.2.5 show the cable assembly and LEDs for the dual-channel PCI card.

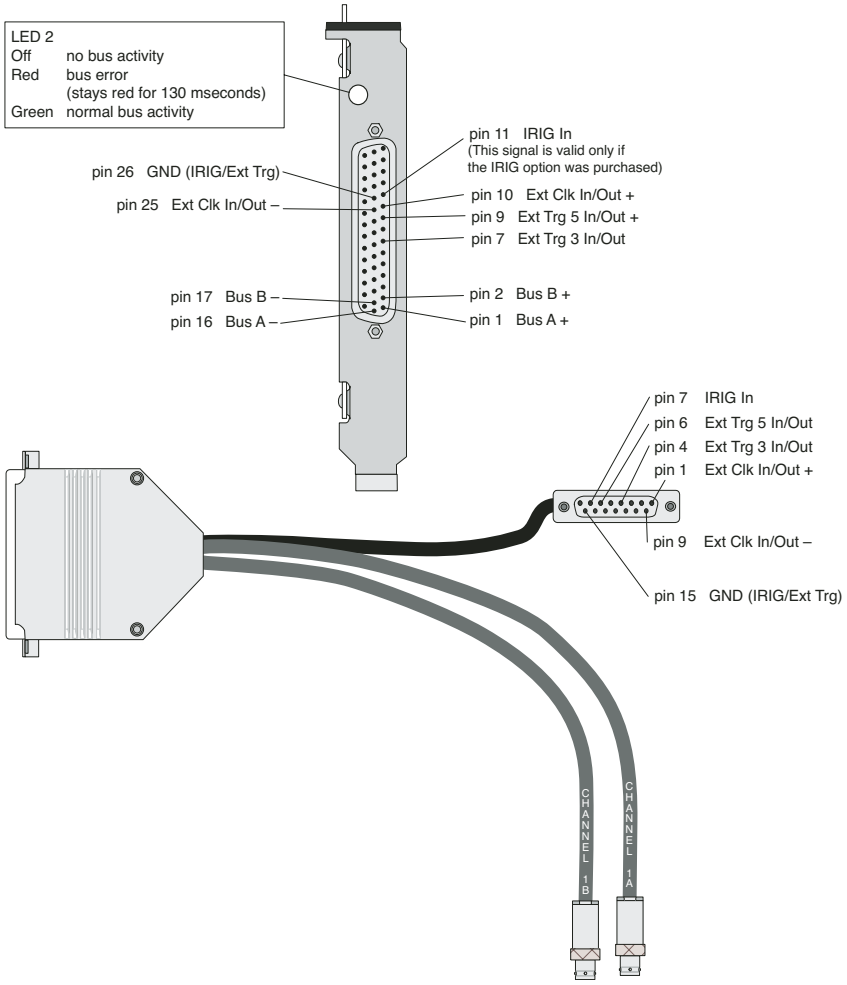


Figure 9.2.2: PCI-1 Cable Assembly and Rear Panel Pinouts and LED

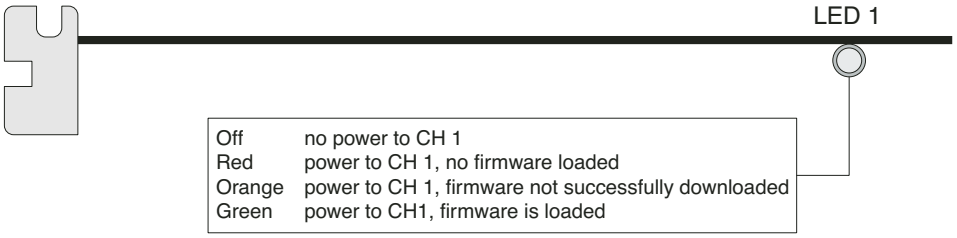


Figure 9.2.3: PCI-1 LED (Top View as Installed)

Notes:

1. External Clock is a differential signal specification.
   + Pin: Rising Edge
   – Pin: Falling Edge

2. External Trigger In/Out is an active low signal with a single-ended specification.

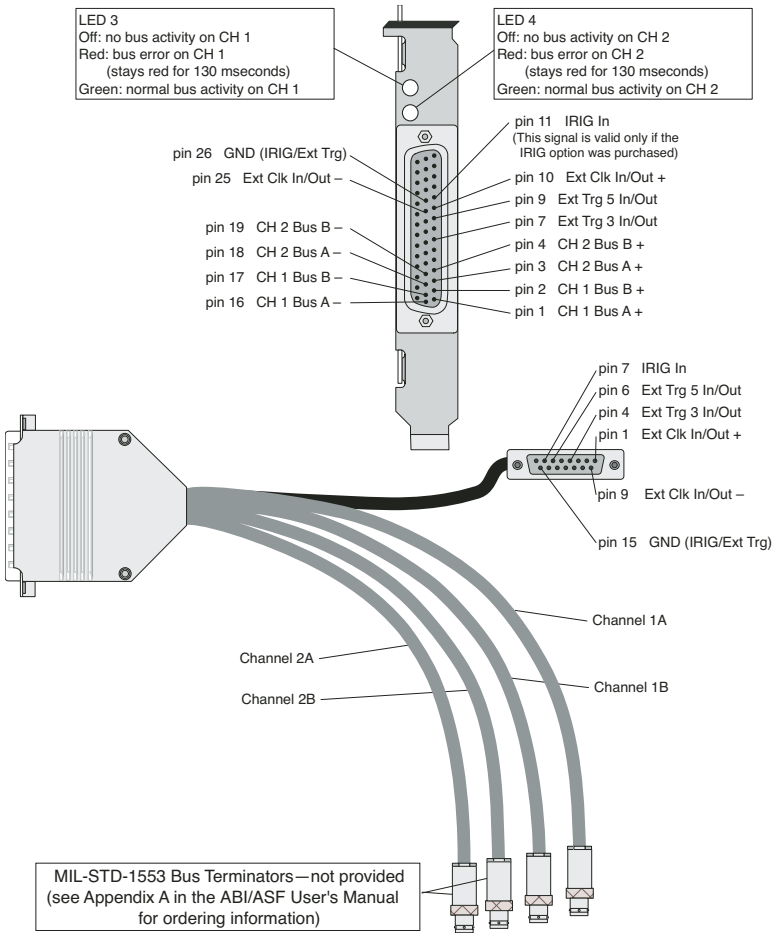


Figure 9.2.4: PCI-2 Cable Assembly and Rear Panel Pinouts and LEDs

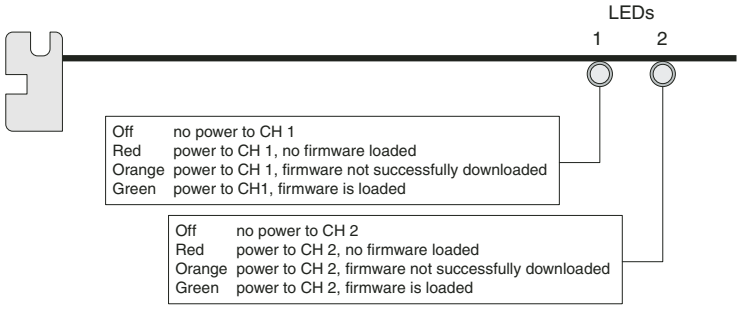


Figure 9.2.5: PCI-2 LEDs (Top View as Installed)

Notes:
1. External Clock is a differential signal specification.
   + Pin: Rising Edge
   – Pin: Falling Edge

2. External Trigger In/Out is an active low signal with a single-ended specification.

9.2.5 Part Numbers and Ordering Instructions

SBS provides one cable assembly, part number CA2088 (single channel), with the PCI. This assembly attaches to the J1 connector on the PCI rear panel. It provides cable leads for making connections to Bus A and Bus B as well as a cable lead with a DB15 connector for making auxiliary connections.

Figure 9.2.6 illustrates the cable assembly and provides part numbers and ordering instructions for all external connectors on the PCI-1. Figure 9.2.7 illustrates the cable assembly and provides part numbers and ordering instructions for all external connectors on the PCI-2.

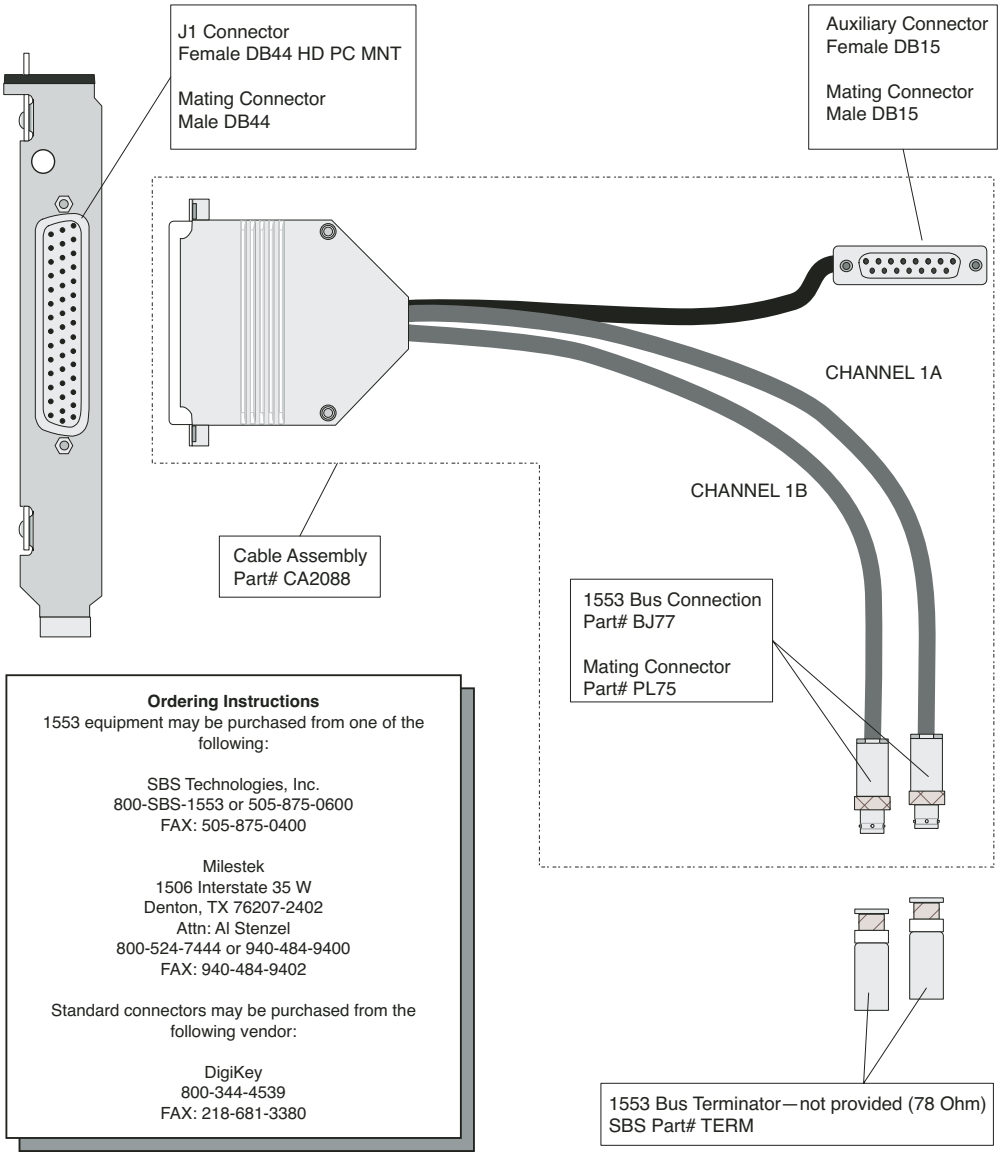


Figure 9.2.6: Part Numbers and Ordering Instructions for PCI-1 Connectors

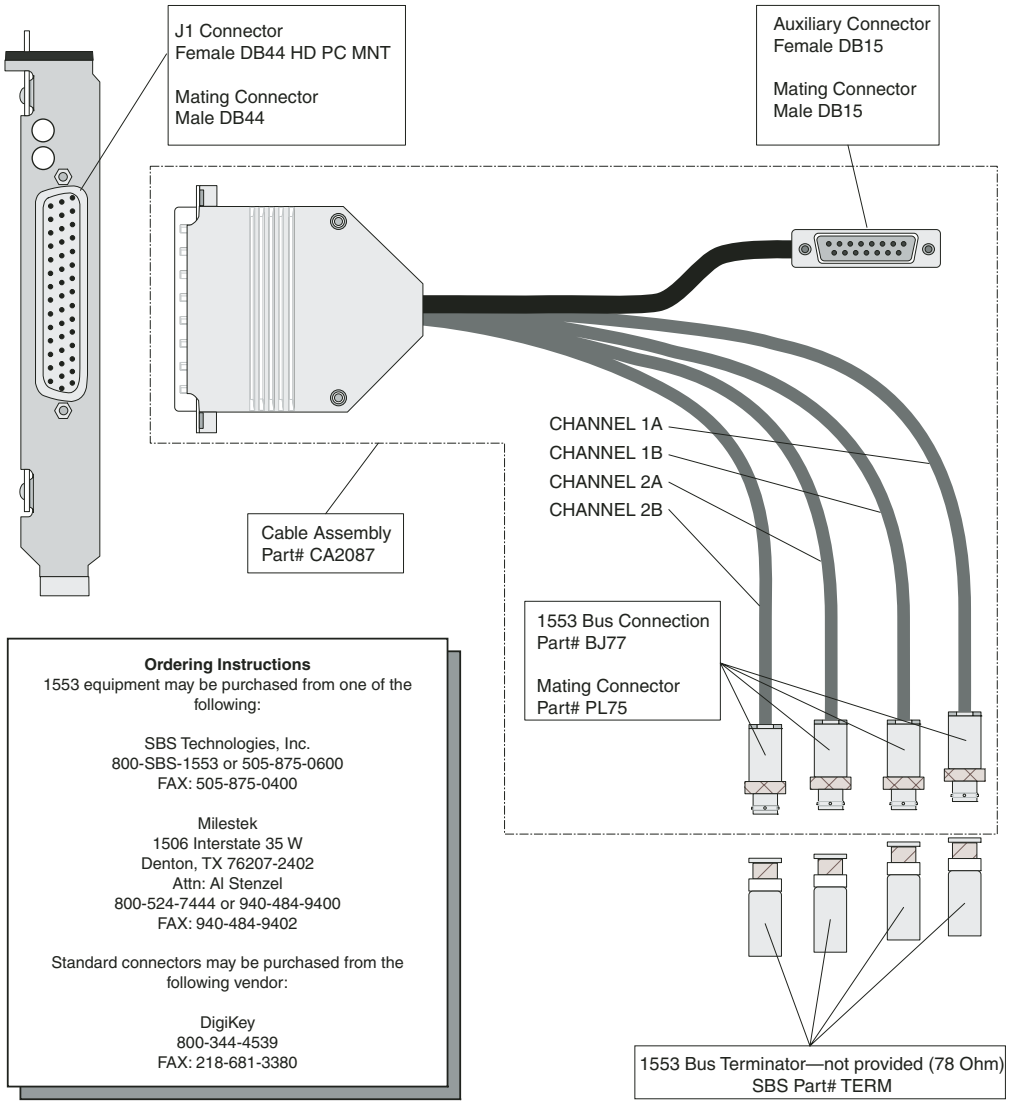


Figure 9.2.7: Part Numbers and Ordering Instructions for PCI-2 Connectors

9.3 Operational Specifications

The information in this section presents the operational specifications for the ABI/ASF-PCI and ABI/ASF-PCI66. This section contains the following topics:

- Temperature
- MTBF
- Hardware Reset

9.3.1 Temperature

Table 9.3.1 lists the PCI operating and storage temperature specifications.

Table 9.3.1: PCI Temperature Specifications

Operating	Storage
0° to +60° C	–65° to +150° C

9.3.2 MTBF

Table 9.3.2 lists the mean time between failures for the PCI. We calculated the MTBF using the MIL-HDBK-217F, Parts Count Method - Ground Benign Environment.

Table 9.3.2: PCI Mean Time Between Failures

PCI-1	PCI-2
95,193 hours	58,892 hours

9.3.3 Hardware Reset

If the hardware reset button on the host system is pressed, the CSR resets and firmware execution halts. At this point, memory above 003Fh is still intact and accessible. After a hardware reset, you must restart the firmware and the memory above 003Fh clears at this time.

Cross Reference: See Section 10.2 for a description of the CSR.

10: Card Information

Note: This manual uses the terms channel and device interchangeably.

This chapter provides card information for the ABI/ASF-PCI cards on memory organization, hardware control registers, and downloading/starting the firmware. This chapter contains the following topics:

10.1 Memory Organization

This section describes the memory organization for the ABI/ASF-PCI card. This section covers the following topics:

10.1.1 Single-Channel Boards

The PCI card reserves one megabyte of memory in PCI memory as shown in Figure 10.1.1.

10.1.2 Dual-Channel Boards

The PCI-2 card reserves two contiguous megabytes of memory in the PCI memory space. Channel 1 reserves the first megabyte. Channel 2 reserves the second megabyte and is offset exactly 80,000h words from Channel 1. Each channel is completely independent and mapped as shown in Figure 10.1.2.

Unless otherwise specified, all memory/register offsets shown in this manual are for Channel 1. To access the respective location in Channel 2, you must add 80000h (word) to the offset. When you use the Integrated Avionics Library which SBS provides, it automatically accounts for the offset to Channel 2. If you directly access the card without using the Integrated Avionics Library, you must provide the offset to Channel 2.

The Software Control Registers and the user-defined firmware structure locations for BC, RT, monitoring, and interrupt handling are between word offsets 0800h and F7FFh. The Control/Status Register (CSR) location is at 20000h with the region from 20001h to 7FFFFh reserved.

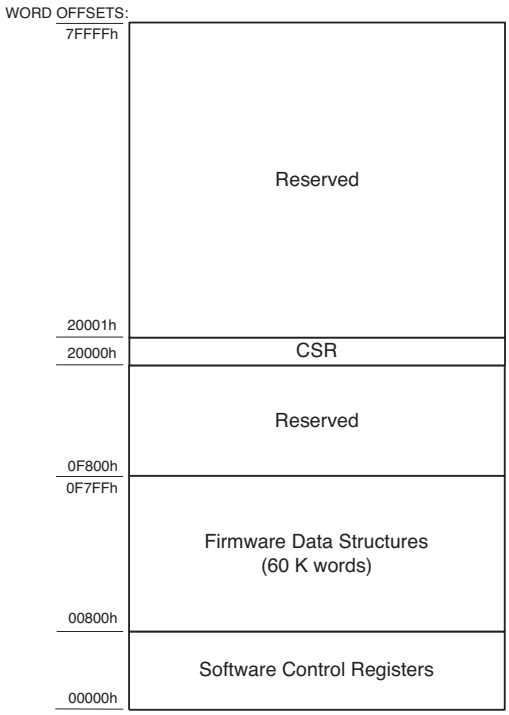


Figure 10.1.1: ABI/ASF-PCI-1 Memory Map

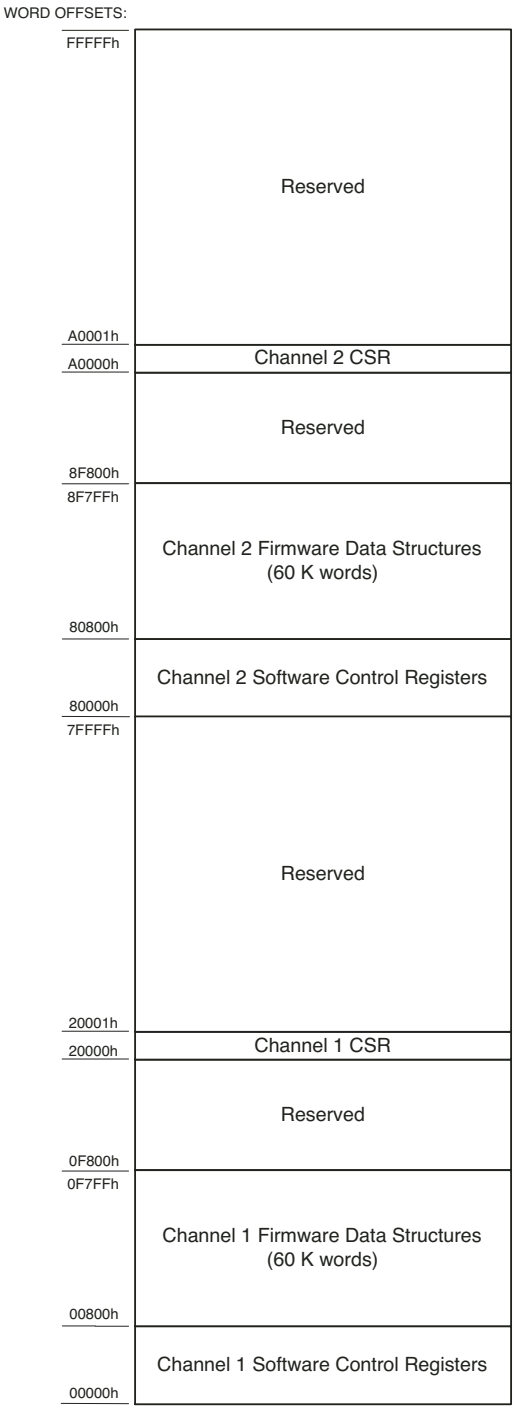


Figure 10.1.2: ABI/ASF-PCI-2 Memory Map

10.2 Control/Status Register

The Control/Status Register (CSR) is a key register for proper initialization and operation of the PCI. Table 10.2.1 describes the bits in the Control/Status Register for Channel 1 of the PCI card (word address 20000h, byte address 40000h), and Table 10.2.2 presents the Control/Status Register for Channel 2 (word address A0000h, byte address 140000h). These registers provide the following module functions to the host:

- Module reset and operation control
- PCI interrupt control

Table 10.2.1: PCI CSR Bit Descriptions (Channel 1)

Bit	Function*	Description
0	Run	0 = Firmware stop 1 = Firmware run
1	DSP Startup Mode	0 = Load from data RAM 1 = Load from FLASH memory
2	Reserved	–
3	PCI Interrupt Enable	0 = PCI Interrupts disabled 1 = PCI Interrupts enabled
4	Reserved	–
5	Enable Word Swap	0 = Word Swap Disabled 1 = Word Swap Enabled
	Note: Word swap is only applicable to the PMC cards. For all other cards, Bit 5 is reserved.	
6	Reserved	–
7	Interrupt Pending (ro) Interrupt Clear (wo)	0 = No interrupt 1 = Interrupt pending (read)/clear interrupt (write) Read this bit to determine whether a host interrupt is pending. Set this bit to clear the interrupt.

*Except where otherwise noted, the host has both read and write access to the CSR bits.
ro = Read only access
wo = Write only access

Table 10.2.2: PCI CSR Bit Descriptions (Channel 2)

Bit	Function*	Description
0	Run	0 = Firmware stop 1 = Firmware run
1	DSP Startup Mode	0 = Load from data RAM 1 = Load from FLASH memory
2	Reserved	–
3	PCI Interrupt Enable	0 = PCI Interrupts disabled 1 = PCI Interrupts enabled
4–5	Reserved	–
6	CH1 Interrupt Pending (ro) CH1 Interrupt Clear (wo)	0 = No CH1 interrupt 1 = CH1 interrupt pending (read)/clear CH1 interrupt (write) Read this bit to determine whether a host interrupt is pending. Set this bit to clear the interrupt.
7	CH2 Interrupt Pending (ro) CH2 Interrupt Clear (wo)	0 = No CH2 interrupt 1 = CH2 interrupt pending (read)/clear CH2 interrupt (write) Read this bit to determine whether a host interrupt is pending. Set this bit to clear the interrupt.

*Except where otherwise noted, the host has both read and write access to the CSR bits.
ro = Read only access
wo = Write only access

10.3 Downloading the Firmware File

The PCI requires all firmware code be downloaded prior to initializing application data structures. A PCI download file contains the firmware code which is provided in ASCII format.

The data in the download file has a 16-bit word structure. The first sixteen words make up the file header, containing product and version information. The seventeenth word in the file contains a word count value (N) for the first half of the data in the file. See Figure 10.3.1.

Complete the following instructions to download the firmware file to the PCI:

1. Open the firmware code file (text mode).
2. Using the I/O Control/Status Register, verify that the internal processor is not running.
3. Read and skip the first sixteen words in the ASCII file.
4. Read the seventeenth word (N). This is the number of data words in the first half of the file.
5. Starting at offset 00800h, read the next word from the file and write the word to memory.
6. Continue reading the file, writing the data, and incrementing the address until N words have been read and written.
7. After the Nth word is processed, repeat steps 4 and 5 with a starting offset of 02C00h until the end of the file is reached.

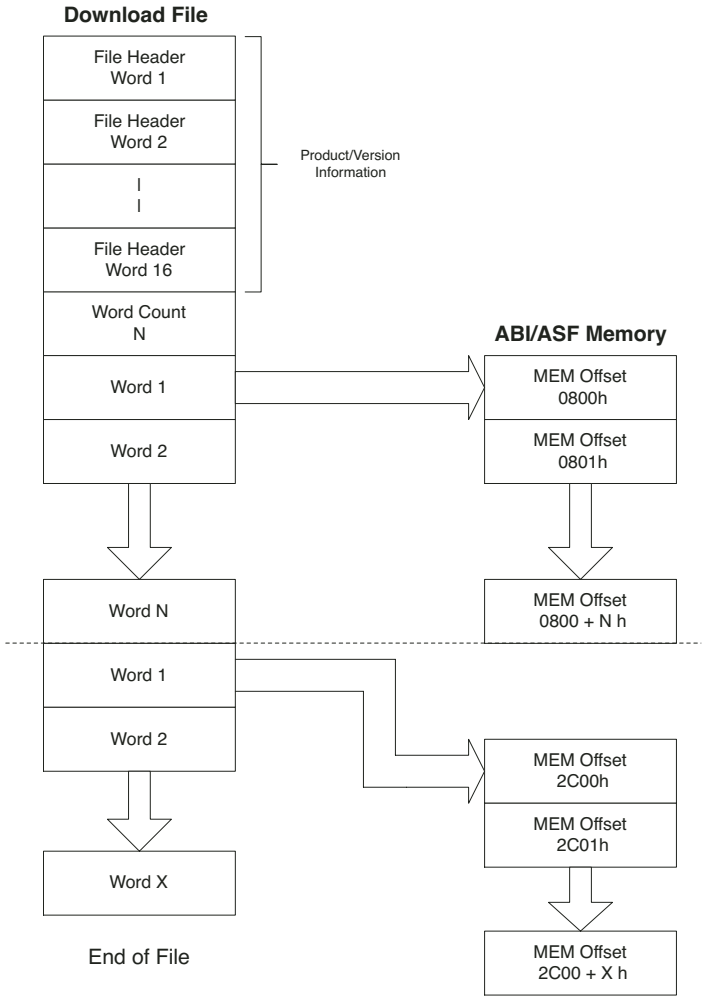


Figure 10.3.1: PCI Download File Format

10.4 Starting the Firmware from On-Board RAM

After powering up or resetting the PCI module, first download the code/data file per the firmware download instructions. Upon completion of the download, perform the following procedure to start up the PCI:

1. Write 0000h to the CSR (offset 0) to reset the PCI hardware.
2. Write 000Ah to offset 07FFh.
3. Write FFFFh to the BIT Status register (offset 3Bh).
4. Write 0001h to the CSR to start the PCI firmware.
5. Read the BIT Status register and wait for the value to equal 0000h, indicating that the power-up tests have completed.
6. Read the BIT total error count (offset 3Ch). The value will be nonzero if errors were detected. If this occurs, contact SBS Technologies' Technical Support.

After completing this procedure, the PCI is in BIT mode awaiting a command. Either select BIT tests to run BIT or initialize the board for 1553 operations.

10.5 PCI Big Endian Access and Data Word Swapping

This section presents the PCI big endian access as used with the ABI/ASF-PCI cards. It covers the following topics:

- Background on Big and Little Endian Data Buses
- PCI Card Memory Architecture
- Big Endian Accesses

10.5.1 Background on Big and Little Endian Data Buses

The PCI bus is an example of a Little Endian bus. On a Little Endian bus, data is long-word aligned to the lowermost byte lane. Byte 0 (address 0) appears in AD[07:00], Byte 1 appears in AD[15:08], Byte 2 appears in AD[23:16], Byte 3 appears in AD[31:24]. The example below shows Little Endian byte ordering using the data pattern 87654321:

Byte	3	2	1	0
Data	87	65	43	21

The Big Endian bus presents the bytes in reverse order. The same data pattern presented in Big Endian byte ordering would look as follows:

Byte	0	1	2	3
Data	21	43	65	87

10.5.2 PCI Card Memory Architecture

The PCI card utilizes a 16-bit DSP processor for 1553 processing. Since a 16-bit DSP is used, the memory architecture also uses a 16-bit base. To facilitate getting data to and from the host at the fastest rate possible, the PCI card provides a 32-bit (long-word) interface between the host and the card's memory. The transition between 16-bit memory and the 32-bit PCI interface raises issues in data byte and word ordering. Table 10.5.1 shows how the data is stored in the card's memory. When the host performs word-based accesses, it reads data in this format.

Table 10.5.1: PCI Memory Data Structure

Word Address	Data
0	4321
1	8765
2	CBA9
3	0FED

When the host performs long-word based accesses, it reads data as shown in Table 10.5.2.

Table 10.5.2: Little Endian Long-Word Access

Long-Word Address	Data
0	87654321
1	0FEDCBA9

While this is desirable in Intel and other Little Endian buses, it may not be desirable in Big Endian systems. In some types of Big Endian systems, it may be desirable to read the data in Big Endian fashion with the bytes reversed as shown in Table 10.5.3.

Table 10.5.3: Big Endian Long-Word Access

Long-Word Address	Data
0	21436587
1	A9CBED0F

In word-based Big Endian systems, it may even be desirable to read data with the words swapped as shown in Table 10.5.4.

Table 10.5.4: Word Swapped Long-Word Access

Long-Word Address	Data
0	43218765
1	CBA90FED

10.5.3 Big Endian Accesses

The PCI also provides Big Endian accesses with the bytes accessed in reverse order. The PLX PCI interface chip provides this capability. Setting bits in the PLX chip control registers activates this capability. When activated, all accesses to the PCI card memory are Big Endian.

Note: Big Endian access is standard on PCI cards.

Interesting Side-Note: The term "endian" comes from a passage in Jonathan Swift's *Gulliver's Travels*.

A: Revisions

The table in this appendix gives a brief summary of any technical revisions made to this manual. When reading this manual online, you can jump to the first citation of a revision by clicking the links in blue.

Note: Only technical revisions appear in the table. Most even numbered pages contain a date stamp in the footer. If the footer date is more recent then the latest revision date given in the table, then the newest revision of this manual contains only *non-technical* revisions.

Revision Number	Revision Date	Description
2	20 Apr 2004	Added ABI/ASF-PCI66 card information. The ABI/ASF-PCI66 card is first cited in the Manual Overview section of Chapter 1. This revision also incorporates all previous revisions.
2.01	20 Oct 2004	Added Uninstalling SBS Drivers In Windows 2000 to Chapter 5.